1992-12

Numerical field model simulation of full scale fire tests in a closed and an open compartment

Delaney, Michael A.

Monterey, California. Naval Postgraduate School

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

1a. REPORT SECURITY CLASSIFICATION Unclassified	1b. RESTRICTIVE MARKINGS
2a. SECURITY CLASSIFICATION AUTHORITY	3. DISTRIBUTION/AVAILABILITY OF REPORT Approved for public release; distribution is unlimited.
2b. DECLASSIFICATION/DOWNGRADING SCHEDULE	
4. PERFORMING ORGANIZATION REPORT NUMBER(S)	5. MONITORING ORGANIZATION REPORT NUMBER(S)
6a. NAME OF PERFORMING ORGANIZATION Naval Postgraduate School 6b. OFFICE SYMBOL *(If applicable)* ME 570	7a. NAME OF MONITORING ORGANIZATION Naval Postgraduate School
6c. ADDRESS *(City, State, and ZIP Code)* Monterey, CA 93943-5000	7b. ADDRESS *(City, State, and ZIP Code)* Monterey, CA 93943-5000
8a. NAME OF FUNDING/SPONSORING ORGANIZATION 8b. OFFICE SYMBOL *(If applicable)*	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER
8c. ADDRESS *(City, State, and ZIP Code)*	10. SOURCE OF FUNDING NUMBERS Program Element No. / Project No. / Task No. / Work Unit Accession Number

11. TITLE *(Include Security Classification)*

NUMERICAL FIELD MODEL SIMULATION OF FULL SCALE FIRE TESTS IN A CLOSED AND AN OPEN COMPARTMENT

12. PERSONAL AUTHOR(S) Michael A. Delaney

13a. TYPE OF REPORT	13b. TIME COVERED	14. DATE OF REPORT *(year, month, day)*	15. PAGE COUNT
Master's Thesis	From To	December 1992	192

16. SUPPLEMENTARY NOTATION

The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

17. COSATI CODES

FIELD	GROUP	SUBGROUP

18. SUBJECT TERMS *(continue on reverse if necessary and identify by block number)*

FIRE MODELING, COMPARTMENT FIRE SIMULATION, COMPUTER MODELING

19. ABSTRACT *(continue on reverse if necessary and identify by block number)*

The intent of this study is to adapt a general numerical model of f ires within an enclosure to a specific experimental test facility. The test facility is operated at the Naval Air Warfare Center, China Lake California. The research facility is in the process of studying missile fuel fires in a surface ship combatant setting. The numerical method employed is capable of generating temperature, velocity, pressure, and density fields for a three dimensional rectangular enclosure. This study is considered preliminary in that much of it concerns the treatment of various difficulties encountered in trying to model a fire with a solid rocket propellant energy release rate using the given computer code. The specific objectives of this thesis are to subject the model to a very high heat input rate and derive results for both a closed and an open compartment. To analyze the open compartment it was necessary to modify the code to incorporate a vent in one of the enclosure walls. A National Center for Atmospheric Research graphics program is used to present isotherm and velocity distribution results.

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT ☒ UNCLASSIFIED/UNLIMITED ☐ SAME AS REPORT ☐ DTIC USERS	21. ABSTRACT SECURITY CLASSIFICATION Unclassified
22a. NAME OF RESPONSIBLE INDIVIDUAL M.D. Kelleher	22b. TELEPHONE *(Include Area code)* 408-646-2530 22c. OFFICE SYMBOL ME/kk

DD FORM 1473, 84 MAR — 83 APR edition may be used until exhausted. All other editions are obsolete — SECURITY CLASSIFICATION OF THIS PAGE

UNCLASSIFIED

NUMERICAL FIELD MODEL SIMULATION
of
FULL SCALE FIRE TESTS
IN A CLOSED AND AN OPEN COMPARTMENT

by

Michael A. Delaney
Lieutenant Commander, United States Navy
B.S., State University of New York at Oneonta,1978

Submitted in partial fulfillment
of the requirements for the degree of

MASTER OF SCIENCE IN ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL
December 1992

ABSTRACT

The intent of this study is to adapt a general numerical model of fires within an enclosure to a specific experimental test facility. The test facility is operated at the Naval Air Warfare Center China Lake, California. The research facility is in the process of studying missile fuel fires in a surface ship combatant setting. The numerical method employed is capable of generating temperature, velocity, pressure, and density fields for a three dimensional rectangular enclosure. This study is considered preliminary in that much of it concerns the treatment of various difficulties encountered in trying to model a fire with a solid rocket propellant energy release rate using the given computer code. The specific objectives of this thesis are to subject the model to a very high heat input rate and derive results for both a closed and open compartment. To analyze the open compartment it was necessary to modify the code to incorporate a vent in one of the enclosure walls. A National Center for Atmospheric Research graphics program is used to present isotherm and velocity distribution results.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

I. INTRODUCTION

A. BACKGROUND

The study of fire propagation through computer modeling has been the subject of on going research at the Naval Postgraduate School (NPGS). There are numerous factors that support the need for this type of research. Some of the more important among them are:

- reduce the need for full scale fire testing
- provide a means for initial analysis of a given scenario so that the actual experiment may be more effectively conducted
- serve to indicate areas within the test compartment that monitoring equipment would be better placed.
- provide for a better understanding of fire propagation and consequently improve firefighting techniques
- allow for a quicker appraisal of a given fire scenario

Although considerable progress has been achieved in this effort, there are clearly good reasons to expand it. It is to this end that this work is undertaken.

The present objective of this research is to develop a numerical procedure to model a fire research test facility operated at the Naval Air Warfare Center, China Lake, California. This research facility is presently conducting tests involving the combustion of missile fuels within a compartment. The facility is designed to mimic the layout and construction of a group of surface ship compartments. The research is motivated by the experiences of shipboard personnel during the Falkland Islands conflict and the Persian Gulf USS Stark incident. Missile fuel fires almost immediately ignited all materials within a compartment causing nearly instantaneous flashover. The facility is the first to employ an empirical study to ascertain the hazards associated with missile propellant combustion in a vented shipboard compartment. Eventually the empirical data recorded at the facility will serve to validate the numerical analysis.

The NPGS effort to model the China Lake facility began with a general numerical procedure designed to model fires within a closed rectangular enclosure. The basic numerical model was developed by H.Q. Yang and K.T. Yang of the University of Notre Dame, Department of Aerospace and Mechanical Engineering. It is entitled "Three Dimensional Numerical Simulation of a Fire Spread Inside a Building" and is provided in Appendix B. The code provided by Professor K.T. Yang was rearranged during a prior thesis but this rearrangement was one of appearance and not of substance. Thus this basic numerical method served as the starting point for the analysis presented in this thesis.

B. OBJECTIVES

The fundamental objective of this thesis is to provide a preliminary assessment of the given numerical procedure's capability in providing solutions for temperature, velocity, pressure and density fields given an extremely potent heat source. No output had previously been generated from this particular numerical method for a heat source characteristic of a solid rocket fuel. Emphasis is placed on the procedure's performance given this type of source and not on the modeling of the source itself. In addition an attempt is made to incorporate provisions for a hole in one of the enclosure walls opening the compartment to the ambient air.

C. PREVIOUS WORK

The program listed in appendix B is a complex compilation of ideas and methods that have developed over a number of years. A variety of sources had to be referred to in order to apply the program to the given problem and analyze the results. A summary of the material (aside from prior thesis work at NPGS) that was found to be helpful is provided here.

Among the most useful references was the book by Patankar [Ref. 1] which served to provide an overall understanding of the basic solution procedure. Doria et. al. [Refs. 2,3] provides background and the specific derivation of equations that were used in a program written at the University of Notre Dame that was a forerunner to the code used in this thesis. The material written by Yang et. al. [Refs. 4-9] and Chan et. al. [Ref. 10] provides a broad background of information concerning both laminar and turbulent flow in various types of compartments. These articles directly address some of the methods applied in the program of appendix B.

Models and procedures which merit considerable discussion in this thesis are a global pressure correction routine, stability criteria, and the application of boundary conditions. The global pressure correction routine is discussed in some detail by Nicollette et. al. [Ref. 11]. Leonard [Ref. 12] has written numerous articles concerning the accuracy and stability of the primary finite differencing technique used in this program. Lin et. al. [Ref. 13] provides specific information concerning boundary conditions for an open channel, unsteady flow type problem and served as an example of that which is considered herein.

An indispensable source in this analysis was the information compiled by Minkowycz et. al. [Ref. 14] from numerous authors concerning numerical methods in heat transfer. Information derived from Greenspan [Ref. 15] and Smith [Ref. 16] was helpful in the application of boundary conditions and the analysis of the general properties of numerical methods. Schlichting [Ref. 17] was a good reference source for the underlying theory concerning fluid motion and governing equations. The radiation heat transfer model's theoretical basis is explained in books such as the one by Sparrow and Cess [Ref. 18]. The Naval Air Warfare Center, China Lake, CA [Ref. 19] provided information concerning the test facility layout, compartment dimensions and the general nature of the tests being conducted there.

D. PRIOR NPGS THESIS WORK

Other useful sources of information utilized in analyzing the numerical model provided in appendix B, are prior theses, [Refs. 20-24], written at the Naval Postgraduate School. Most of this work concerns the modeling of the FIRE-1 test facility, at the Naval Research Laboratory which is designed to provide empirical data for fires in a cylindrical compartment. It was developed to simulate submarine compartment fires. Nies [Ref. 20] provides an in depth description of the basic three dimensional numerical model for fire propagation in a closed enclosure. Finite difference equations, turbulence models, radiation models, and conduction models are developed based on a Cartesian coordinate system. Most of the derivations discussed by Nies are directly applicable to the model provided in appendix B. Raycraft [Ref. 21] applied equations based on spherical and cylindrical coordinate systems to more accurately model the FIRE-1 test facility. A more detailed formulation of the radiation model is also provided. Houck [Ref. 22] used the spherical and cylindrically based geometry of the Raycraft formulation and provided an internal ventilation model. McCarthy [Ref. 23] applied the existing computer code to the FIRE-1 test facility and used CA-DISSPLA Graphics to present data generated by the program.

Work on the modeling of the China Lake test facility was begun by Thorkildsen [Ref. 24]. Some preliminary runs of the program were accomplished and a NCAR Graphics program was developed to analyze the isotherm and velocity profiles generated by the numerical model.

The major difference between the analysis conducted in this thesis from previous work is that the data presented in the previous theses are for a comparatively small fire. The velocity, pressure, density, and temperature gradients were not nearly of the magnitude generated by the type of fire being studied at the China Lake facility. In addition, no analysis was previously accomplished for an open compartment.

E. PREVIEW OF SUBSEQUENT CHAPTERS

Section B of Chapter I provides the layout of the China Lake facility along with some information of the instrumentation available there. Chapter II presents information related to the output of results by the program. It is intended to provide a basis for discussion of these results. It was necessary to address the numerical properties of the main finite difference scheme employed, the conduction model, boundary conditions, and the global pressure correction routine. An explanation of the modifications made to generate the enclosure opening in the program are also provided in

Chapter II. The third chapter concerns the analysis of data generated by the numerical scheme. NCAR Graphics is used extensively to aid in analyzing the data. Chapter IV is a summary of the research conducted and provides some conclusions and recommendations. Appendix A provides a general program flow chart.

F. CHINA LAKE FACILITY DESCRIPTION

An isometric and plan view of the ship compartment simulator at the Naval Weapon Center (NWC), China Lake is given in figures 1.1 and 1.2 respectively. The facility consists of a main compartment (20ft by 20ft by 10ft high), a compartment (15ft by 15ft by 10ft high) adjacent to the main compartment and a compartment (15ft by 15ft by 10ft high) mounted above the main compartment. All compartments are constructed with 3/8 inch thick steel bulkheads and 1/2 inch thick steel decks. The bulkheads are structurally reinforced with 5 inch I-beams and the deck plating is reinforced with 12 inch I-beams. There are no partitions within the compartments.

Prior to a test run, a ventilation opening of a specific size is incorporated into the wall opposite that of the fire location in the main chamber in accordance with the test scenario. The vent is intended to simulate a missile impact hole in the hull. The fuel pan is placed on the south side of the main chamber enclosure approximately four feet from the east wall. Wooden cribs of various sizes are placed in the compartment when called for by the test scenario to simulate class 'A' fire materials.

Instrumentation is provided to measure fire compartment gas temperatures, temperatures of escaping gases, air temperatures in the adjacent and overhead compartments and total heat flux at various locations. The bulk compartment pressure can also be measured in the main chamber's northeast corner. Portable thermocouple trees can be installed at any desired location. They extend from 1/2 foot off the deck to 1/2 foot from the compartment overhead.

The tests are run with various amounts of solid rocket propellant in a fuel pan and range from 25 to 200 lbs. The combustion rate for each test was determined by the length of time that it took for the propellant to completely dissipate. From the data made available to the Naval Postgraduate School it ranged from 0.9 to 4.5 lbm/s. Following discussions with test facility personnel, a heat of combustion of 2600 BTU/lbm was estimated for the fuel.

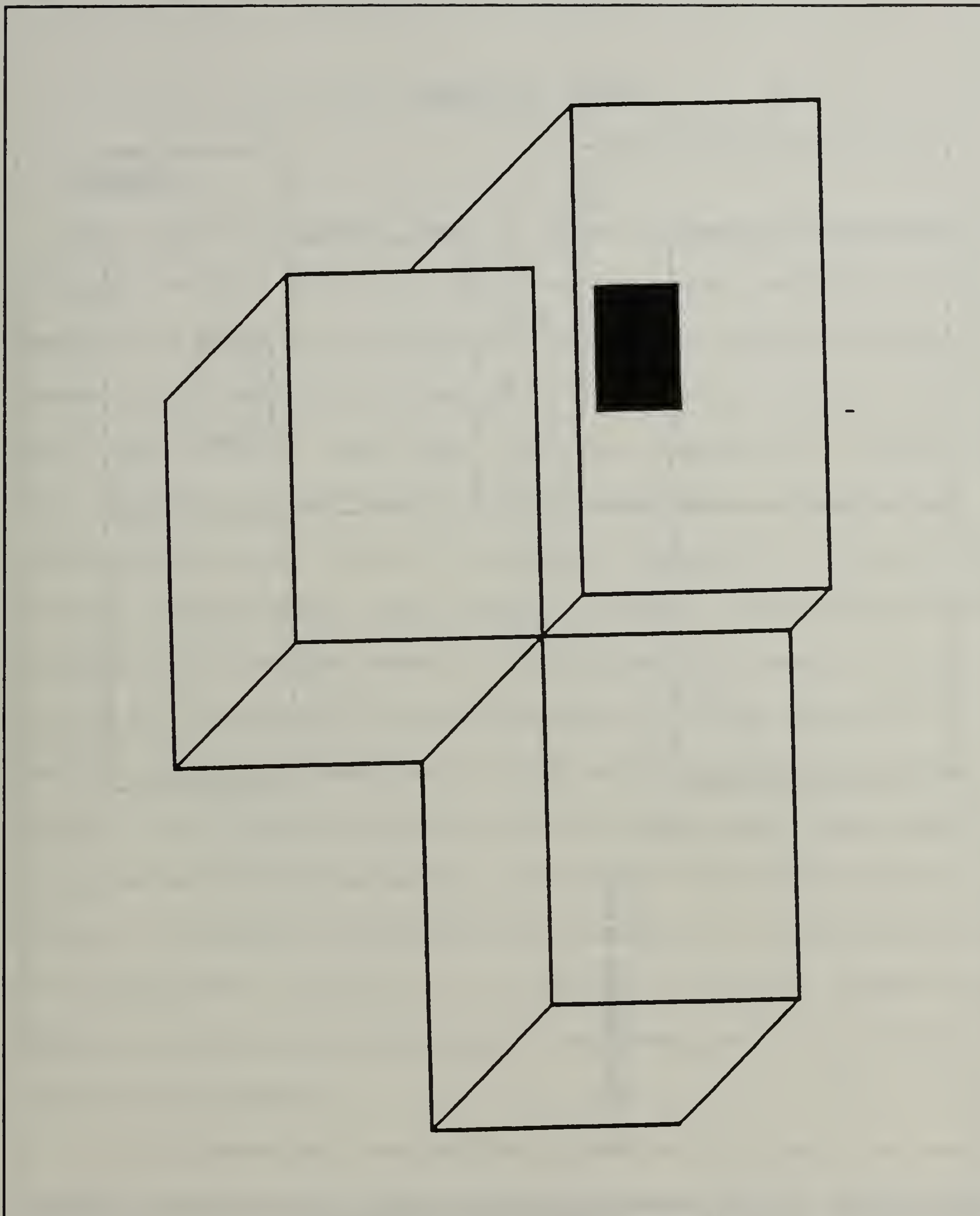


Figure 1.1 Fire test chamber at NAWC, China Lake, CA.

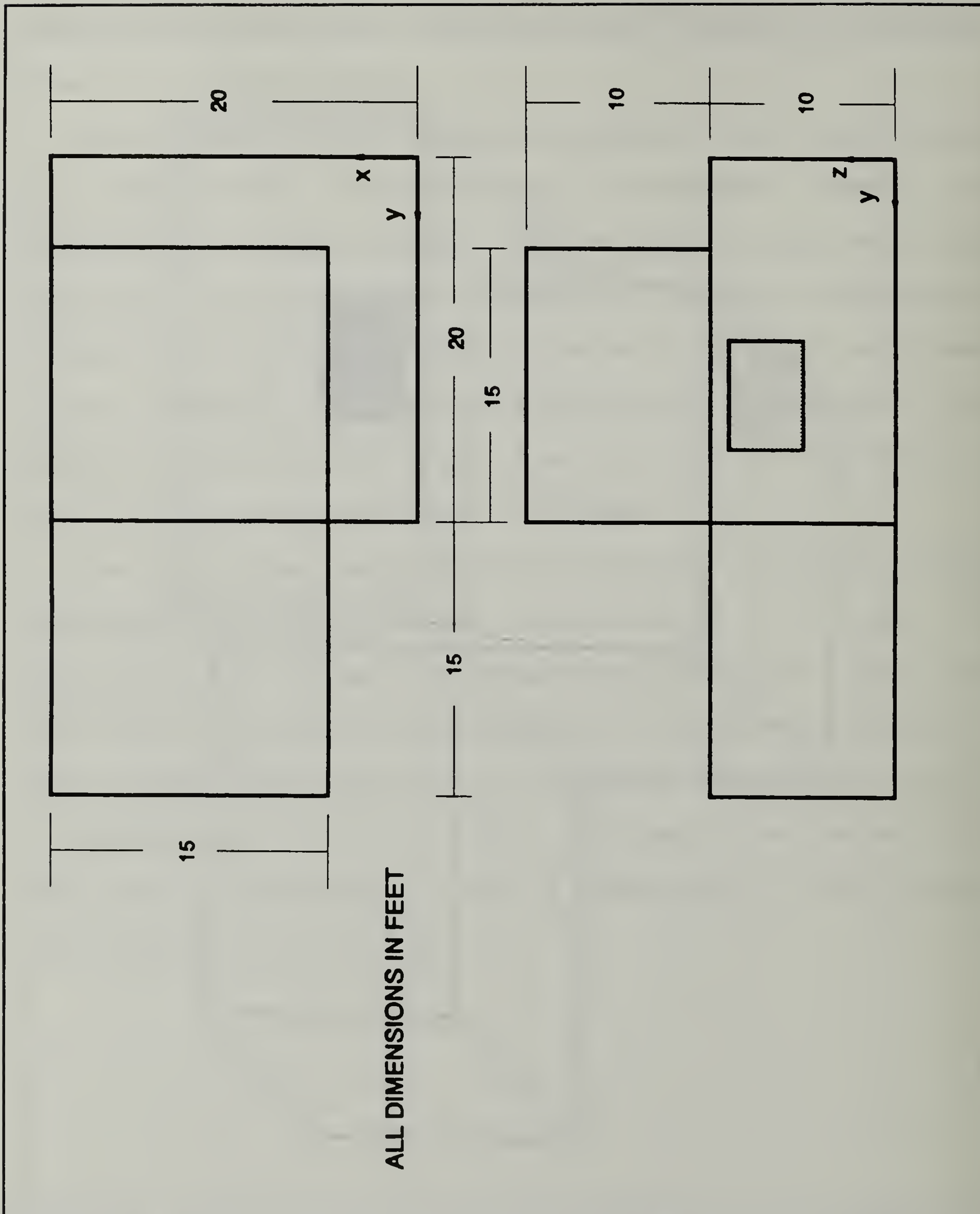


Figure 1.2 Plan view of fire test chamber

II. NUMERICAL MODEL

A. GENERAL

The numerical method used in this analysis incorporates different models for the transport of mass, momentum, and temperature which are linked in a sequential order to develop temperature, velocity, pressure, and density fields for a given heat source. The model for the compartment interior (area within the enclosure walls) uses three dimensional, primitive variable, finite difference methods to calculate velocity, temperature, and pressure fields. The perfect gas law equation of state serves to calculate the density field. It is also incorporated into conservation of mass equations to provide a global pressure correction which assists in solving for the true pressure field in the compartment. The model incorporates strong buoyancy, turbulence and compressibility effects. Provision is made for the placement of solids within the compartment. Finally, a tridiagonal matrix algorithm (TDMA) is used to solve the equations generated by the finite differencing method.

A one dimensional conduction model is provided for heat transfer calculations through the enclosure walls. Enclosure walls of different thermodynamic properties may be input separately into the program.

The heat source may be modeled by any appropriate algorithm and distributed into any portion of the compartment desired. At present provision is made for only a uniform distribution of the source into the control volumes chosen for its location. A radiative heat transfer model quantifies radiative heat transfer from the heat source to the enclosure walls and solids located in the compartment interior. An important feature of the program is that it allows the use of a nonuniform grid.

A detailed development of the basic numerical method in cartesian coordinates is provided by Nies [Ref. 20]. However, the finite differencing technique described by Nies for the space within the enclosure walls is the UPWIND scheme whereas the program used in this analysis utilizes the QUICK (quadratic upwind interpolation for convective kinematics) method which was developed by Leonard [Ref. 12]. The derivation of the finite difference equations for a general coordinate system using the QUICK method is given by Raycraft [Ref. 21] and is directly applicable to this particular aspect of the program. In addition, the conduction model described by Nies is not the conduction model used in this program.

The use of the general numerical method described by Nies and Raycraft to generate results for a very high energy heat source and the incorporation of a vent opening in one of the enclosure walls poses several difficulties. In section B of this chapter the model of the compartment interior is addressed with emphasis on the numerical properties of the QUICK scheme. Because problems are encountered in using the program to generate results for large magnitude fires, it is necessary to establish some idea of the factors affecting grid spacing and time steps. Section C addresses the specific conduction model used in this program. This model appears to have a significant impact on the results presented in Chapter III. Sections D and E discuss boundary conditions and the global pressure correction routine which are major considerations in modifying the program to incorporate the vent opening.

B. MODEL OF THE COMPARTMENT INTERIOR

The governing equations for a three dimensional buoyant flow are well known and are given by Schlichting [Ref. 17]. The equations are reduced to dimensionless form by utilizing the following definitions:

$$x_i = \frac{\tilde{x}_i}{H}, \quad u_i = \frac{\tilde{u}_i}{u_R}, \quad p = \frac{\tilde{p}}{\rho_R} u_R^2$$

$$T = \frac{\tilde{T}}{T_R}, \quad t = \frac{\tilde{t} u_R}{H}, \quad g_i = \frac{\tilde{g}_i H}{u^2{}_R} \qquad \text{eqn. } 2.1$$

$$\rho = \frac{\tilde{\rho}}{\rho_R}, \quad C_{pm} = \frac{\tilde{C}_{pm}}{C_{pR}}, \quad \mu = \frac{\tilde{\mu}}{\rho_R u_R H}$$

$$k = \frac{\tilde{k}}{\rho_R C_{pR} u_R H}$$

where the quantities denoted with the tilde symbol are and i=1,2,3. Thus $\tilde{x}_i$ denotes the cartesian coordinates x,y,z and H is the height of the compartment. $\tilde{t}$ is dimensional time, $\tilde{g}_i$ is the component of the gravitational acceleration in the i direction, $\tilde{u}_i$ the velocity vector, $\tilde{T}$ the temperature, $\tilde{p}$ the static pressure, $\tilde{\rho}$ the fluid density, $\tilde{C}_{pm}$ the fluid mean specific heat, $\tilde{\mu}$ the fluid viscosity and $\tilde{k}$ the fluid thermal conductivity. The governing equations in tensor notation using the above definitions can be written as follows:

CONTINUITY:

$$\rho_t + (\rho u_i)_{,i} = 0 \qquad \text{eqn. 2.2}$$

MOMENTUM:

$$(\rho u_i)_{,t} + (\rho u_i u_j)_{,j} = -p_{,i} - \rho g_i + (\sigma_{ij})_{,j} \qquad \text{eqn. 2.3}$$

ENERGY:

$$(\rho c_{pm} T)_{,t} + (\rho u_i c_{pm} T)_{,i} = (kT_{,i})_{,i} + \mu\Phi + Pu_{i,i} \qquad \text{eqn. 2.4}$$

where the t subscript denotes derivatives with respect to time. The dimensionless shear stress tensor σ_{ij}, c_{pm} and the dissipation function Φ are given respectively by:

$$\sigma_{ij} = \mu(u_{i,j} + u_{j,i} - \frac{2}{3}\delta_{ij} u_{k,k})$$

$$c_{pm}=\frac{1}{T-1}\int_{1}^{T}c_p\,dT \qquad \text{eqn. 2.5}$$

$$\Phi=2(u^2{}_{i,j})\delta_{ij} + [u_{i,j}(1-\delta_{ij})]^2 - \frac{2}{3}(u_{i,i})^2$$

where δ_{ij} is the Kronecker delta function. It should be noted here that the program neglects the viscous dissipation.

The basic numerical method is developed using the control volume approach described by Patankar [Ref. 1]. As stated in Ref. [1], the general form of the energy, momentum and continuity equation is:

$$\frac{\partial}{\partial t}[\rho\phi] + \nabla\cdot(\rho\vec{u}\phi)=\nabla\cdot(\Gamma\nabla\phi) + S \qquad \text{eqn. 2.6}$$

where ϕ is the dependent variable. The diffusion coefficient Γ, and **S** correspond to the specific meaning of ϕ. The discretized form of eqn. 1 may in general be expressed as:

$$A_p\phi=A_E\phi_E+A_W\phi_W+A_N\phi_N+A_S\phi_S+A_F\phi_F+A_B\phi_B+S \qquad \text{eqn. 2.7}$$

where the subscript P refers to the center of the control volume and the subscripts E,W,N,S,F,B refer to quantities determined for points at the center of the East,West, North, South, Forward, and Bottom neighboring control volumes of P. S can be viewed as a source term and a collection of quantities that do not neatly fit into the coefficients for the center and neighboring grid points. The coefficients A_p through A_B and S are determined by the specific differencing scheme used.

The approach espoused by Patankar utilizes the 'centered grid' to calculate the dependent variables of temperature, pressure and density and the 'staggered grid' to calculate each component of velocity. In effect, the temperature, pressure and density calculations yield solutions for discrete points at the center of each centered control volume and the calculations for the three components of velocity give solutions for discrete points located at the face of each centered control volume. The energy equation is used to determine the temperature field. The momentum equations are coupled to the continuity equation to derive the velocity field and the pressure field. This is done because of the difficulty in solving for the pressure gradient which appears in the momentum equation. This particular procedure is clearly explained in Refs. 1 and 14. The continuity equation actually functions as an extension of the momentum equation to determine a pressure correction for each control volume. This is termed the 'local' pressure correction. These local pressure corrections are then used to both complete the velocity calculation and update the value for pressure at the nodes to which they apply. The perfect gas equation of state is used to derive the density field.

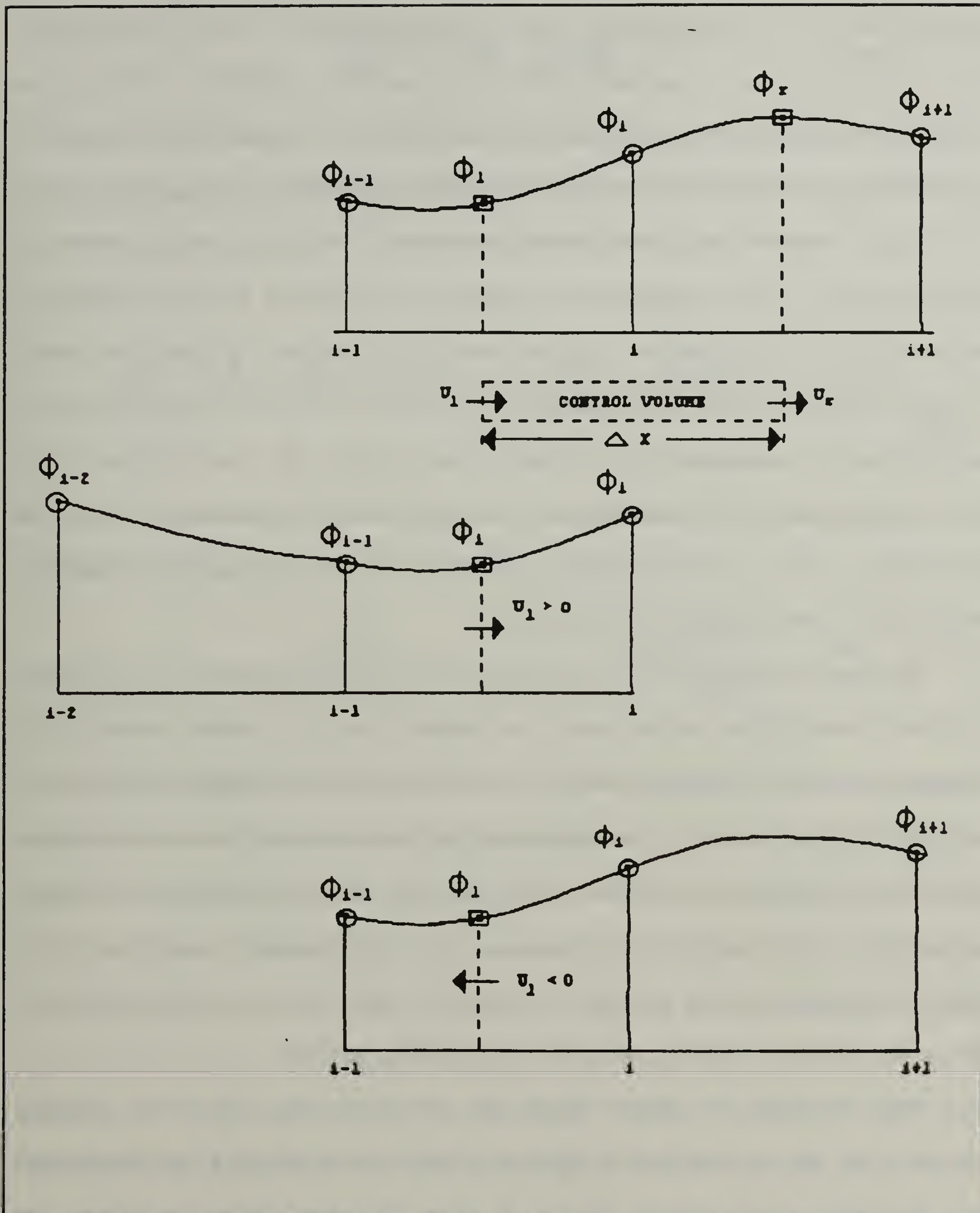


Figure 2.1 QUICK Scheme Portrayal

The pressure field which results from the method described above is not, in general, correct as pointed out by Nicolette et. al. [Ref. 11]. As explained by Patankar [Refs. 1,14], the local pressure correction is derived by demanding that it satisfy the continuity equation and eliminate the source term for each control volume. This provides for the local pressure variation in the compartment. However, it does not provide for determining any changes to the overall (global) pressure level in the space. Since the density calculations are dependent on an accurate assessment of the true value of the pressure at each node, an additional correction for the pressure field is required. This is provided by the global pressure routine which is described in section E.

The application of the QUICK finite difference technique to the governing equations is described in great detail by Raycraft and Houck [Refs 21,22]. Their formulation is explained in terms of a general coordinate system and can be applied directly to the derivation of the discretization equations used in this program. The derivation need not be repeated here. More apropos to this thesis is a discussion of the numerical properties of the QUICK method.

The method is described in general by Leonard [Refs. 12,14]. It is an explicit finite difference algorithm designed for highly convective flow. A one dimensional approach is employed for analyzing stability constraints and truncation errors. For one dimension and constant density, eqn. 2.6 reduces to the following:

$$\frac{\partial \phi}{\partial t} = -\frac{\partial u\phi}{\partial x} + \frac{\partial}{\partial x}\Gamma\frac{\partial \phi}{\partial x} + S \qquad \text{eqn. 2.8}$$

where symbols have the same meaning as described for eqn 2.6. After integration this equation can be expressed as shown by Leonard [Ref. 14] and repeated here for convenience.

$$\phi_{i+1}^{n+1} = \phi_i^n + c_l^n\phi_l^n - c_r^n\phi_r^n + [\alpha_r^n(\frac{\partial \phi}{\partial x})_r^n - \alpha_l^n(\frac{\partial \phi}{\partial x})_l^n] + \Delta t S_i \qquad \text{eqn. 2.9}$$

where c represents the Courant number and α represents the diffusion parameter:

$$c = u\frac{\Delta t}{\Delta x} \qquad \alpha = \Gamma\frac{\Delta t}{\Delta x} \qquad \text{eqn. 2.10}$$

Figure 2.1 serves to illustrate the meaning of the subscripts. r denotes the interface to the right of point i and l to the left. S represents the space time averaged source term. The goal of the numerical interpolation scheme is then to accurately calculate the values for ϕ and its gradient at the interfaces. As the name implies, the QUICK method uses a quadratic interpolation involving three nodes, the two immediately adjacent to the interface and then the next node upstream depending on the direction of the velocity at the interface. Dropping the superscript n, this results in the following expressions for ϕ:

$$\phi_l=\frac{1}{2}(\phi_i+\phi_{i-1})-\frac{1}{8}CURV\Delta x^2 \qquad \text{eqn. 2.11}$$

where

$$CURV=\frac{\phi_i-2\phi_{i-1}+\phi_{i-2}}{\Delta x^2} \qquad u_l>0$$

$$CURV=\frac{\phi_{i+1}-2\phi_i+\phi_{i-1}}{\Delta x^2} \qquad u_l<0 \qquad \text{eqn. 2.12}$$

The expression for the gradient term is equivalent to a central differencing between the points i-1 and i:

$$\left(\frac{\partial\phi}{\partial x}\right)_l=\frac{\phi_i-\phi_{i-1}}{\Delta x} \qquad \text{eqn. 2.13}$$

The formulation for the right face is similar and can be correctly displayed by adding the number one to each of the subscripts and changing the l to an r in eqns. 2.11-2.14.

Because the gradient term is expressed by what amounts to a central differencing technique, the QUICK scheme is formally second order ($O(\Delta x)^2$) in the spatial variable [Refs 14,16]. However for conditions in which the convective transport is dominant over diffusive transport, the scheme becomes essentially third order accurate in the spatial variable. shown by Leonard [Ref. 12], a Taylor series expansion around i-1 reveals that:

$$\phi_l=\phi_{l(exact)}+\frac{1}{16}\left(\frac{\partial^3\phi}{\partial x^3}\right)_i\Delta x_l^3+H.O.T \qquad \text{eqn. 2.14}$$

The method is explicit and the truncation error in time is of

$(O(\Delta t))$ [Refs. 14,16]. The method is therefore not really suited for particularly unsteady behavior.

The stability constraints on the time step are described by Leonard [Ref. 14] using a Fourier-von Neumann analysis of a one dimensional, source free finite difference equation for constant values of u and Γ. The solution for ϕ is considered to be of the form:

$$\phi = A(t)\, e^{ik\Delta x} \qquad \text{eqn. 2.15}$$

where k is the wave number. Eqn 2.15 is substituted into eqn 2.9 and an expression for the ratio of the amplitudes (A(t+1)/A(t)) is found. Furthermore a cell Peclet number is defined as $P_{cell}=c/\alpha$. The result is that the restrictions on the time step can be expressed as:

$$\Delta t \le \frac{2\Delta x^2}{4\Gamma + u\Delta x} \qquad \text{eqn. 2.16}$$

and

$$c \le \frac{2}{P_{cell}} + \frac{\pi^2}{2N^2} \qquad \text{eqn. 2.17}$$

where N is the number of grid subdivisions in the given direction. These restrictions correspond to the low convection and high convection regimes respectively. The restriction given in eqn. 2.17 is actually an approximation to the formal Von Neumann criteria. The Von Neumann method places no restrictions on the size of the domain and considers waves of infinite length as part of the solution for a given variable. The resulting expression for the time step is much smaller. In actual practice a grid size is finite and only waves of a finite length will enter into the solution. The second restriction results from this more realistic approach.

Although the entire discussion was for a one dimensional case, it is not inapplicable to the three dimensional code used here. This is because the one dimensional QUICK scheme is applied individually to each dimension to determine values of the given variable at the faces of the control volume. It is therefore reasonable to assume that the code should be subject to the basic limitations described by Leonard. In applying the analysis in Chapter III, each dimension was analyzed and the time calculated by considering the most restrictive case and reducing by a factor of three. The application of eqns. 2.16 and 2.17 serve only as a basic estimate of the time step required for the given grid configuration. At the very least, the above discussion serves to define the relevant parameters affecting the stability and accuracy of the scheme.

Some mention of the impact of the implementation of boundary conditions on the stability and accuracy of the differencing scheme must also be made. For the open compartment, the boundary conditions are set by what amounts to a forward differencing scheme which is first order accurate in both the time and spatial variables [Ref. 16]. This has to be considered in designating the time step. Determination of the grid spacing was in accordance with the considerations described by Patankar [Ref. 14]. This amounts to making trial runs and attempting to evaluate where finer grid spacing is required by analyzing the magnitude of the velocity fields developed as well as the solutions for the other dependent variables.

The description of the radiation model and stress calculations which are incorporated into the model of the compartment interior are identical to that given by Raycraft and Houck [Refs. 21,22]. The viscosity and conductivity are calculated using an algebraic model for turbulent flows and appears to be identical to that described by Nies [Ref. 20] and Raycraft [Ref. 22]. Boundary conditions and the global pressure correction routine merited much consideration in modifying this code and are dealt with separately in sections C and D of this chapter.

C. MODEL OF THE ENCLOSURE WALLS

Heat transfer through the enclosure walls is assumed to be one dimensional (normal to the plane of the wall). As can be seen from figure 2.2 , three nodes and the ambient temperature are used to provide the temperatures at the outer face and the internal wall node. The equations used to derive an expression for T_1 are given in eqn. 2.18 where h is the convection coefficient between the wall and the ambient air. It is made a constant and assigned a value indicative of natural convection.

$$q''_{wall}=h(T_1-T_\infty)$$

$$q''_{wall}=-k_2\frac{(T_2-T_1)}{\frac{DXI}{2}} \qquad \text{eqn. 2.18}$$

The equation used to solve for T_2 is:

$$\rho c_{pm}[T_2-T_2^{old}]\frac{Vol}{\Delta t}=k_2\frac{(T_1-T_2)}{DXI}\Delta Z\Delta Y+k_e\frac{(T_3-T_2)}{DXI}\Delta Z\Delta Y+q_r \qquad \text{eqn. 2.19}$$

where k_e is the geometric mean between k_2 and k_3 (the conductivity of the wall and air respectively). q_r is the radiative heat transfer from the fire to the wall control volume. The value of T_2 is used in the finite difference scheme for the compartment interior and the value of T_3 is calculated in the differencing scheme for the interior.

In the radiative heat transfer calculation, the surfaces of the enclosure are considered to be gray so that their absorptivity is equal to their emissivity. Only radiative heat transfer from the fire to the enclosure walls is considered (for the case in which there are no solids present in the compartment) and air is a non participating medium. The vent surface area is modeled as a black body of ambient temperature and the absorptivity is equal to unity. This heat transfer does not play a role in the derivation of T_2. T_2 in the vent area is provided by the boundary conditions. The radiative heat transfer calculation is still made but the results are stored separately and then incorporated into the energy balance for the system.

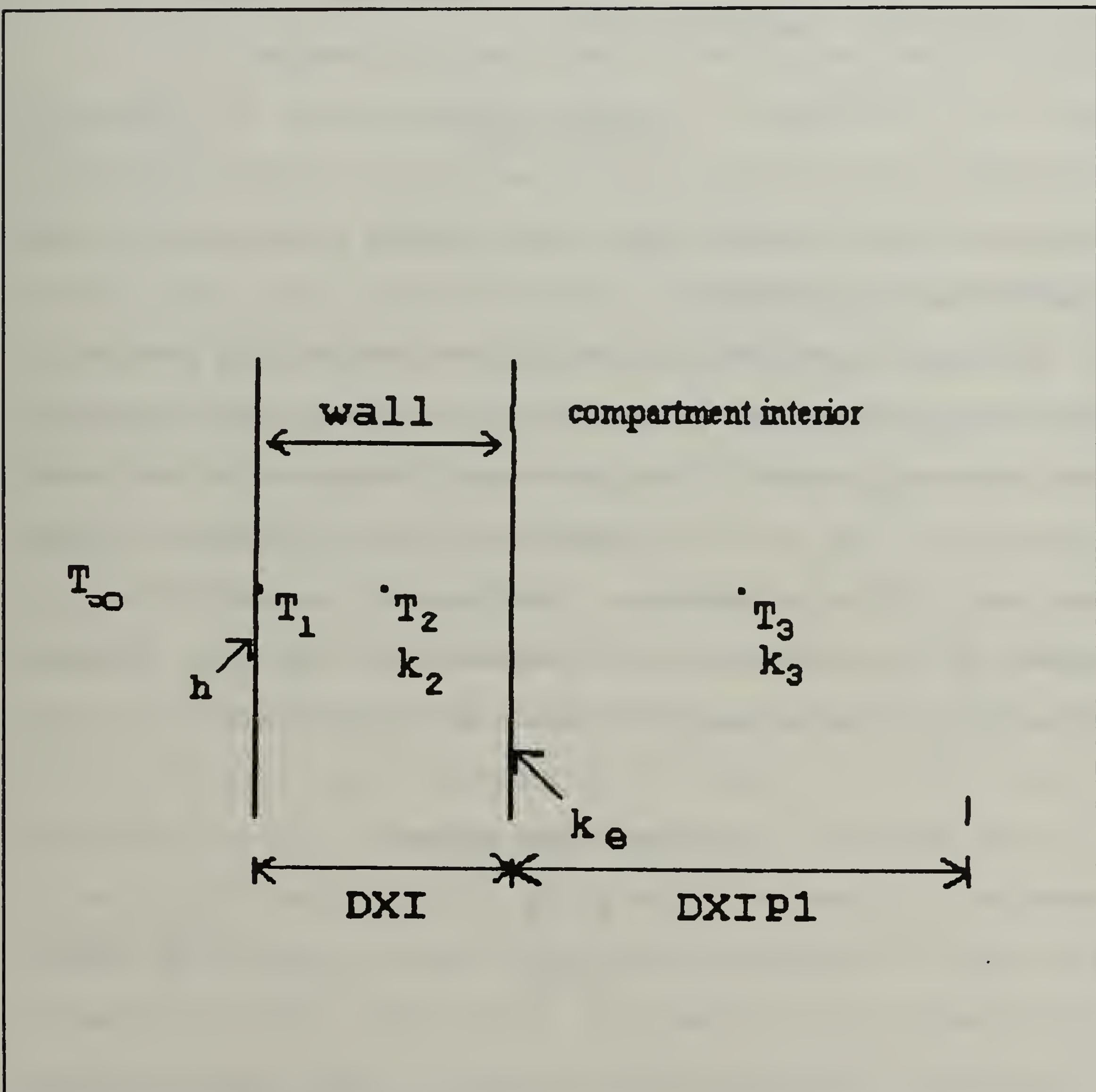


Figure 2.2 Conduction Model

D. INITIAL AND BOUNDARY CONDITIONS

The initial conditions for the model are a quiescent air mass and the entire compartment in equilibrium with the ambient conditions. All velocity components are set to zero. The pressure and temperature is that of the ambient. The equilibrium pressure and density fields are calculated using the hydrostatic approximation:

$$P = P_o \exp - \left(\frac{gz}{RT_A} \right) \qquad \text{eqn. 2.20}$$

Pressure field corrections are applied relative to the equilibrium quantities.

Boundary conditions are complicated somewhat by the use of the staggered grid and the combination of the conduction model with the compartment interior model. Figure 2.3 is a two dimensional view of the boundary and node placement on the west wall of the compartment. The boundary conditions can be summed up by the following statements although they require explanation. The subscript b denotes the boundary.

AT ENCLOSURE WALLS

$$\vec{V}_b = 0$$
$$T_b = T_{amb} \qquad \text{eqn. 2.21}$$

IN THE VENT AREA

$$\left(\frac{\partial \vec{v}}{\partial n}\right)_b = 0$$

$$P_b = P_{amb} \qquad \text{eqn. 2.22}$$

$$T_b = T_{amb} \qquad u_b > 0$$
$$T_b = T_{interior} \qquad u_b < 0$$

1. Closed Compartment

The temperature field for the entire enclosure (walls included) is established through a combination of the conduction model for the walls and the finite difference algorithms for the space within the enclosure walls. As was pointed out in the conduction model discussion, the boundary value used in the finite difference equations is provided by the conduction equations. Therefore the temperature boundary value for the entire system would be the value of temperature of the air outside the compartment.

The momentum equation is coupled to the continuity equation and the conditions for velocity are therefore intertwined with those for the pressure correction. The grid around the west wall is depicted in figure 2.3. The first u components in the interior are located on the inner face of the wall and the calculational control volumes are centered around the column of nodes that are vertically in line with the xs(i+1,j) node. This is not the case for the v components, nodes for which fall on either side of the face of the wall. The method followed in the program is to manipulate the coefficients that are in the form of eqn. 2.7. Referring to figure 2.3, the equation for the v component of velocity is:

$$A_P V_P = A_W V_W + A_E V_E + A_N V_N + A_S V_S + S \qquad \text{eqn 2.23}$$

where the points P,W,E,N,S correspond to points ys(i,j), ys(i+1,j), ys(i-1,j), ys(i,j+1), and ys(i,j-1) respectively of figure 2.3. Although not derived in this thesis, it should be understood that the coefficient for point P includes the coefficients on the right hand side of the equation. Because V_P is in the wall, the coefficients on the right hand side of eqn. 2.23 are set to zero and A_P is assigned an artificially large value which is the semi-implicit method of setting the value of velocities for nodes inside of solids to zero. The no slip condition is implemented for the next line of nodes to the east of ys(i,j). This is approximated by setting the condition V(i+1,j) = - V(i,j). The following manipulation is then made:

$$A_P V_P = A_W V_W + A_E V_E + A_N V_N + A_S V_S$$
$$V_P = -V_W \qquad \text{eqn 2.24}$$
$$(A_P + A_W)\, V_P = A_E V_E + A_N + V_N + A_S V_S$$

where the subscripts P,W,E,N,S now refer to the nodes ys(i+1,j), ys(i,j), ys(i+2,j), ys(i+1,j+1), and ys(i+1,j-1) respectively. For the u component of velocity, the coefficients for the nodes on the inner wall face such as xs(i+1,j) are treated in the same manner as in equation 2.23. The coefficients for the next line of nodes to the east are dealt with in the manner described by eqn. 2.24.

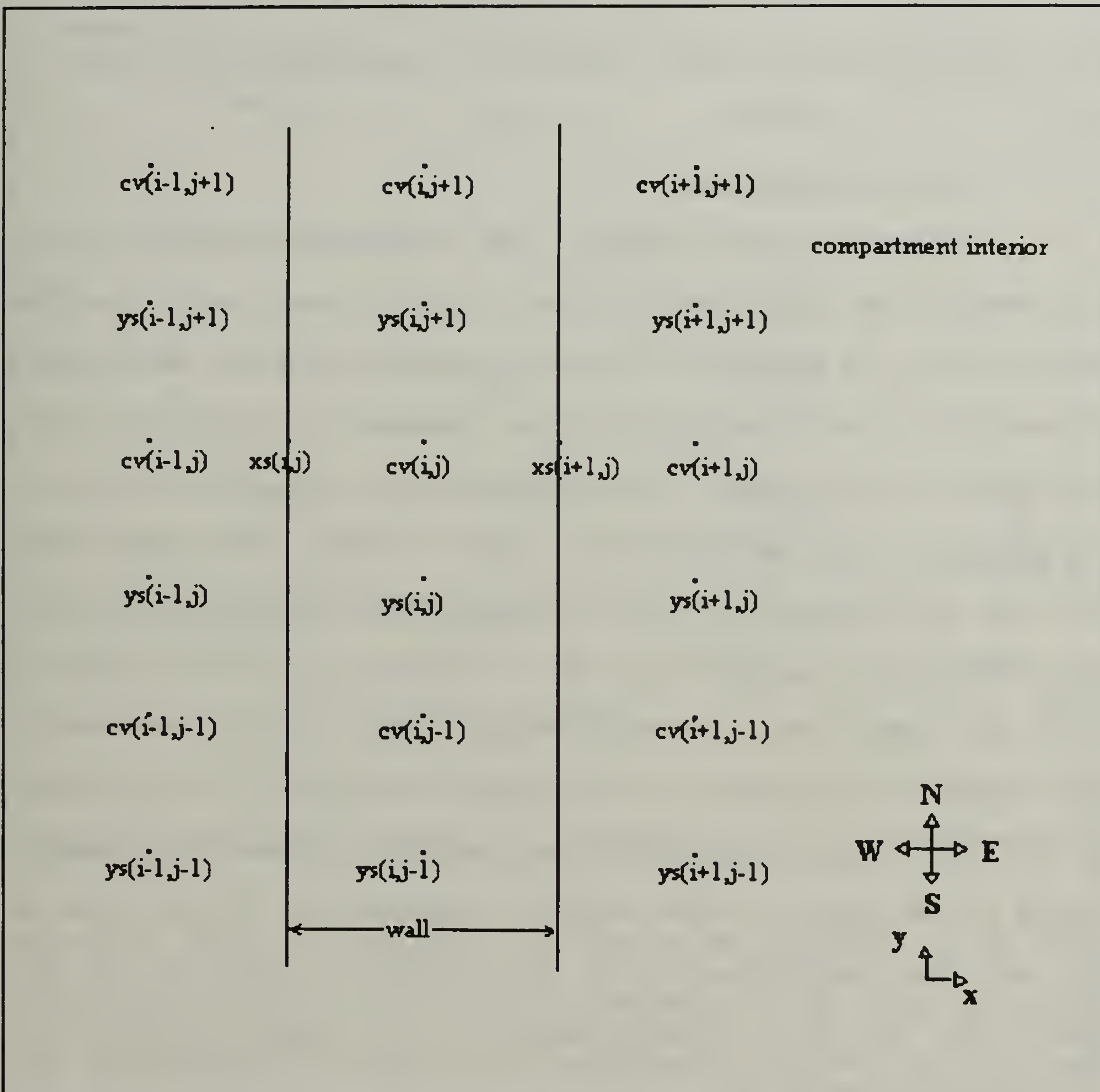


Figure 2.3 Boundary conditions at walls

The pressure correction calculational control volumes are centered about the cv nodes (refer to figure 2.3). For the pressure correction calculations, the coefficients of interior compartment nodes which involve nodes in the wall are dealt with using the semi-implicit method. The A_w coefficients for the first line of nodes inside the compartment are set to zero.

2. Open Compartment

Where a wall exists, the boundary conditions are implemented as described for the closed compartment. For the vented area the boundary value for temperature is designated depending on the direction of the component of velocity which is normal to the vent. The calculational control volumes for temperature are centered about the cv nodes. The nodes that fall on the boundary for the energy equation are those which are vertically in line with B1 in figure 2.4. If the flow is into the compartment (determined by whether the u components of velocity calculated for the nodes vertically in line with B2 in figure 2.4 are positive or negative) then the boundary value is set equal to the ambient temperature. If the flow is out of the compartment then the boundary value for is set equal to the node immediately to the EAST (inside the compartment) of it.

For the vent area the gradient of all components of velocity in the direction normal to the plane of the vent (east-west direction) is set to zero. For the v component of

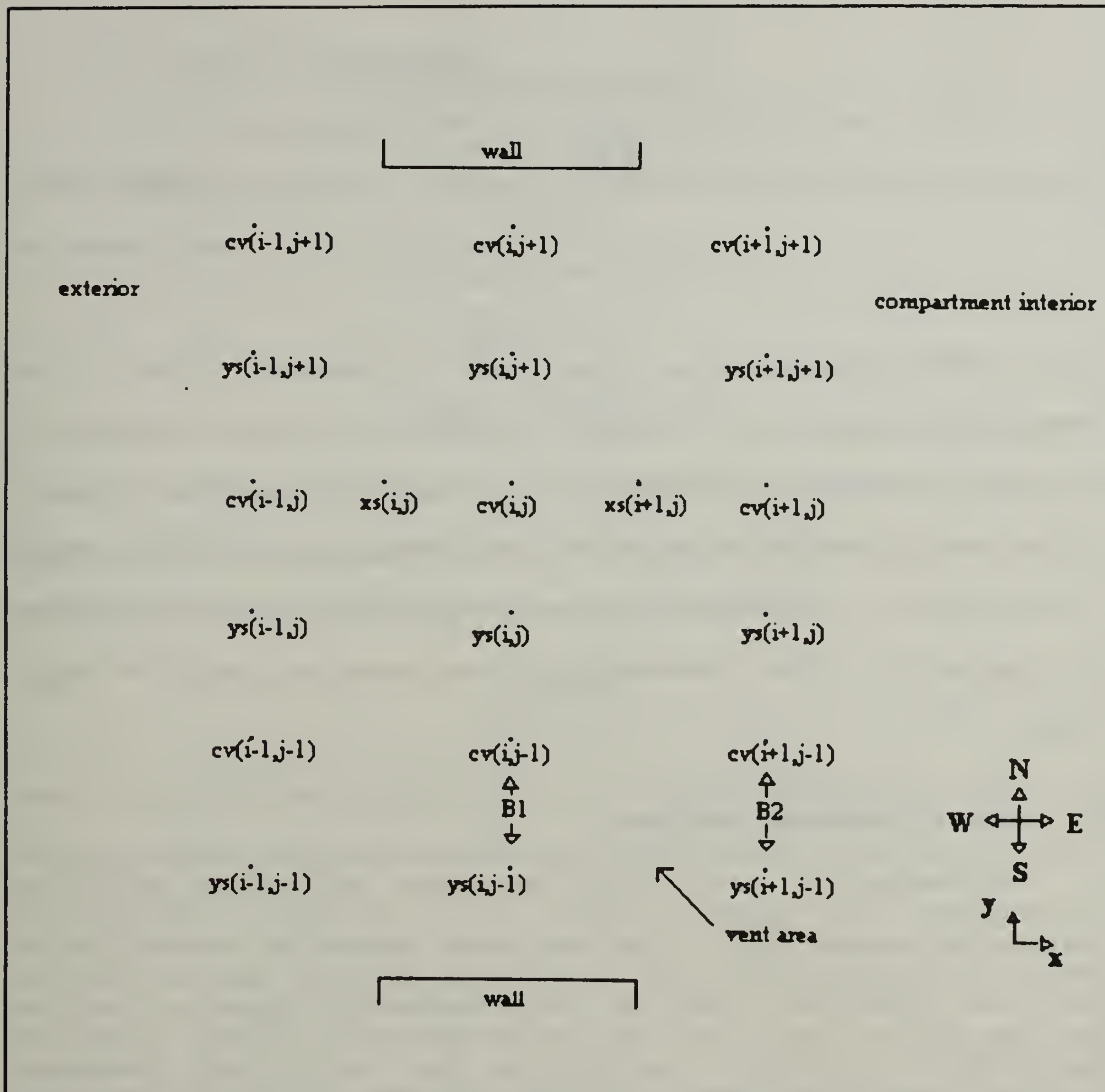


Figure 2.4 Boundary conditions in the vent area

velocity, this corresponds to the nodes vertically in line with B1 in figure 2.4. For the u component, the first nodes in the calculational domain are vertically in line with B2 and therefore the zero gradient specification is applied there. Using the u component for an example, the manipulation of coefficients proceeds as follows:

$$A_P U_P = A_W U_W + A_E U_E + A_N U_N + A_S U_W$$

$$\frac{\Delta U}{\Delta x} = 0 \quad so,$$

eqn. 2.25

$$V_P = V_W$$

$$(A_P - A_W)\, U_P = A_E U_E + A_N U_N + A_S U_S$$

The stipulation that the pressure outside the compartment does not rise requires that no pressure correction be developed for the first node in the vent area that is part of the calculational domain for the pressure correction. This corresponds to the nodes that are vertically in line with B1 in figure 2.4. The procedure is the same as that for eqn. 2.24.

E. GLOBAL PRESSURE ROUTINE

The basis for the global pressure correction which is calculated separately from the local pressure correction is that the transient term in the continuity equation can be manipulated to yield a correction for the overall pressure field as described by Nicollette et. al. [Ref. 11]. Nicollette's formulation applied specifically to a closed compartment. The description which follows for the closed compartment parallels that of Nicollette. It was found necessary to try to extend this type of correction to the open compartment based on results observed during this thesis work.

1. Closed Compartment

In the formulation described by Nicollette [Ref. 11], the transient term for a fixed volume-fixed mass system can be expressed as:

$$\frac{\partial}{\partial t}\sum \rho \Delta V = 0, \qquad \text{eqn. 2.26}$$
$$\sum \rho^{(t+\Delta t)} \Delta V = \sum \rho_{eq} \Delta V$$

where V denotes volume. The density is a function of both pressure and temperature. The true value of the pressure and temperature at any point can be written as:

$$P = P* + P' \qquad \text{eqn. 2.27}$$
$$T = T* + T'$$

where the starred quantities are those calculated from the previous time step and the primed quantities are a correction. Nicollette's method is to first ignore the temperature correction and then to manipulate eqn 2.26 using the perfect gas law to provide a pressure correction which is then applied to each cell. The density field is then calculated and the temperature field iterated with the new values of density. The procedure provides for a more accurate assessment of the temperature and density fields prior to solving for the velocity field and local pressure correction.

2. Open Compartment

For the open compartment, the formulation of the global pressure correction routine for the closed compartment is no longer valid. For one of the trial runs presented in Chapter III, the global routine was simply not applied, thinking that the results would not be markedly affected. From the analysis of that trial run, it was apparent that a global correction was required. An attempt was made to account for the mass convected from the compartment opening by modifying eqn. 2.26 as follows:

$$\sum(\rho^t)\Delta V=\sum\rho_{eq}\Delta V+\sum_{0}^{t}(\rho A\vec{v}_n)_{vent}\Delta t \qquad \text{eqn. 2.28}$$

where the vent term represents the mass that has flowed from the vent. The outcome is discussed in Chapter III.

III. NUMERICAL RESULTS

A. BACKGROUND

The objective of the work beginning with Thorkildsen [Ref. 24] was to model a solid rocket propellant fire in a compartment in the enclosure described in chapter one. The results which were presented in Ref. 24 demonstrated the model's effectiveness only in modeling a very mild heat generation rate type fire (approximately 26 BTU/S or 27.4 KW). Instability problems were encountered in trying to increase the heat generation rate and the emphasis in Thorkildsen's work was therefore on the development of graphics programs to display data. These graphics programs were indispensable in analyzing the numerical experiments described herein. The work done by Nies, Raycraft, and Houck [Refs. 20-23] involved modeling of fires generated by fuels with a comparatively small heat of combustion and overall much smaller energy input rate into a compartment. The heat release rate of the type of fire considered here is approximately two orders of magnitude greater than those modeled previously. The results presented herein are considered preliminary in that no claim is made to have accurately modeled the China Lake test facility. The discussion in this presentation centers around the numerical experiments conducted in attempting to make this particular code produce realistic outputs given the magnitude of the energy input into it and the attempt to incorporate an enclosure opening.

The first topic addressed in this chapter is the input of the heat release characteristics of the fire and the thermodynamic properties of the enclosure materials. The next issues that are discussed are the actions taken to prevent instability in the program. This includes the grid and time step utilized. Finally an analysis of the results is presented.

B. HEAT SOURCE AND INPUT PARAMETERS

The development of an algorithm to accurately model the characteristic heat release rate of a solid rocket propellant was not a priority in this thesis. Two factors that were considered were that the fuel is self oxidizing and therefore is relatively unaffected by the surrounding conditions and that the fuel almost instantaneously reaches its maximum burn rate. In view of these facts the burn rate was set for a linear increase until it reached its maximum rate. In Thorkildsen's work [Ref. 24] the rate of the linear increase was set for a period of two seconds. For the runs conducted during this thesis the linear increase took place over the first eight seconds of fire time. The rate that energy was distributed into the compartment was modified by estimating the energy radiated from the source to the walls of the compartment. This was done by summing up the radiation to the walls from the previous time step and subtracting it from the quantity calculated for dispersion into the interior compartment control volumes during the current time step. This was a feature of the program received from Notre Dame.

As a first approximation it was desired to get output for a 1.0 lbm/s (0.45 kg/s) burn rate. It was eventually found that 1.0 lbm/s was too high given the fuel's heat of combustion and burn rate characteristic input into the program. The significant properties input into the program are summarized in the following table.

Table I MATERIAL PROPERTIES SUMMARY

Wall and Deck Properties	
Wall Thickness	0.375 in
Deck Thickness	0.5 in
Specific Heat	0.103 BTU/(lbm F) 0.432 KJ/(kg k)
Thermal Conductivity	63 BTU/(hr ft F) 115 W/(m K)
Type	AISI 1010
Heat Source Properties	
Heat of Combustion	2600 BTU/lbm
Burn rate:	
Trial 1:	1 lbm/s
Trials 2-6:	0.5 lbm/s

Discussion of mass burn rates will henceforth be made in English units. All other quantities will be discussed in SI units.

C. GRID and TIME STEP

The first step taken in attempting to derive output from the program for a large heat release rate was to refine the grid. In the beginning stages of the fire the largest components of velocity are in the vertical direction and the largest gradients are in the overhead and around the fire itself. This could be observed from the plots of the results from previous theses [Refs. 20-24]. Because the program could incorporate a non-uniform grid, it was decided to vary the grid spacing in the vertical direction throughout and then to also modify the grid spacing in the horizontal directions around the fire. Several variations were tried and the eventual outcome was a grid with one half foot spacing in the vertical direction from the floor to the overhead and one half foot spacing in both horizontal directions in the area of the fire. In addition extra nodes were incorporated into the vent area on the west wall. A plan view of the horizontal grid spacing is provided in figure 3.1. The energy input was distributed into a total of 24 control volumes stretching from 16 feet to 18 feet in the x direction, 8 feet to 10 feet in the y direction and 1 foot to 5.5 feet in the z direction. This is an area which is roughly 6 percent of the entire volume of the enclosure. The final result was a grid for the interior of the compartment of the size 25 by 25 by 20. Additional nodes are required for the walls so the total grid used to solve the problem was 30 by 30 by 25. Each run in the analysis that follows was run with this 30 by 30 by 25 nonuniform grid.

The criterion for the time step was outlined in chapter two. This can serve only as a general guide because the local cell Peclet numbers are changing throughout the problem. In addition the diffusion parameters of viscosity and conductivity are variable, the grid is nonuniform, and the treatment of boundary conditions entails greater inaccuracies. What results is a very general estimate indeed. However, choosing to use the parameters that affect the calculation of the momentum equation (kinematic viscosity) and then evaluating eqns. 2.16 and 2.17 results in a time step criterion of less than 0.66 seconds and 0.0026341 seconds respectively. The following parameters were used to arrive at this conclusion:

$$\nu = 1.56E\text{-}4\frac{ft^2}{s}$$
$$u = 1\frac{ft}{s}$$
$$\Delta x = 1\ ft$$
$$N_{(NR\ grid\ points)} = 25$$

In running the numerical experiments it was found that time steps of up to 0.0125 seconds provided consistent results but this depended on the type of problem run. Each run is annotated with the time steps used.

D. RESULTS

The results of each trial are discussed individually. Plots of various quantities are given for similar locations in the compartment for comparison purposes. Only a small amount of information that can be output from the program is included herein. The atmospheric pressure in the compartment varied by less than one percent from the bottom of the compartment to the top for any specific time. This serves to demonstrate the reason why it would be very difficult to find the local pressure variation for use in the momentum equations by using the perfect gas law. The pressure graphs indicate the atmospheric pressure variation in the compartment with time. The isotherm plot and thermocouple units are degrees celsius, the velocity is in cm/s and the pressure plots in atmospheres. The heat input rate to the compartment varied due to the attempt to consider the radiative transfer as explained previously. The heat input rate in kilowatts is therefore given for each trial except the first.

Trial 1 demonstrates some of the stability problems that were encountered in running the program. Trial 2 was a run for a closed compartment. Trial 3 was intended to simulate the open compartment. No global pressure correction was applied in trial three. Because of unexpected results in trial 3, a run was conducted (trial 4) for a closed compartment with no global pressure correction applied. Trials 5 and 6 were run due to observations made in runs 2 and 3. The orientation for viewing the plots is given in figure 3.2.

A note of caution is made concerning the CPU run times given in the analysis. To say that a certain number of CPU seconds are required for a second of fire time is quite inaccurate. The solution procedure is an iterative one - some time steps may require far more computational effort than others. Larger fires are more difficult to solve for than smaller ones. The CPU time listed should therefore be the basis for only a very general comparison between runs. The computer used was the AMDAHL MODEL 5995-700A located at NPGS.

1. Trial 1

In this trial the results from a run in which the burn rate was set to 1.0 lbm/s illustrate some of the difficulties in getting the program to perform for a high energy input level. The velocity and isotherm plots are given in figures 3.5 to 3.7. The intent was to burn approximately 25 lbm of fuel and to record the buildup of the fire and the subsequent post fire phase in an open compartment. The input level was ramped up over a period of eight seconds to its maximum rate of 2745 KW. At the 10 second mark areas in the compartment had already reached temperatures in excess of 4000 C. The fire burn rate was kept at that level until the 25 second mark at which time the energy input was set to zero. Figure 3.6 shows the compartment temperatures exactly at this time. The isotherm indicates that the compartment air mass is almost uniformly above the 2000 degree mark and it has become difficult to locate the fire at that point. Figure 3.7 shows the same elevation view 1.3 seconds later. Most areas are already 2000 degrees cooler, a situation which is not believable. The velocity plot indicates a large change in conditions at the vent area. The excess mass term generated in the continuity equation calculations exceeded its maximum limit at the 26.8 second mark.

One lesson learned from this and other runs including those for a closed compartment with a global pressure correction applied were that the numerical model has difficulty incorporating almost instantaneous changes in energy input level. This is undoubtedly due to the fact that such changes lead to severe gradients and the existing grid refinement becomes inadequate for resolving these gradients.

The other lesson learned was that a burn rate of 1.0 lbm/s appears excessive given the heat of combustion used in these trials. The temperature field reaches levels in excess of 4000 degrees very quickly in both the open and closed compartment. Therefore runs are subsequently conducted at half that burn rate.

2. Trial 2

Trial two was for a closed compartment with a mass burn rate of 0.5 lbm/s for 60 seconds. The time step utilized was 0.003125 seconds for the first 10 seconds, 0.009375 seconds for the period from 10 to 20 seconds and then 0.0125 seconds until completion. The total amount of CPU time required was 29865 seconds. The pressure and heat input curves are presented in figure 3.3. Figures 3.8 through 3.15 present the velocity and isotherm plots. The velocity plots appear to concur with the isotherm plots as the fire progresses. The elevation view for Y equal to 8 feet (figure 3.8) which cuts directly through the fire shows the upward plume growing stronger with time. Other velocity plots show a circulation pattern around the compartment. By 60 seconds the entire air mass above the midlevel of the compartment is above 1000 C. The fire location is becoming more difficult to pinpoint from views above the compartment midlevel. A much colder and mostly stagnant air mass can be seen at the bottom of the compartment. This gradually becomes a much smaller percentage of the compartment as time progresses.

One of the most glaring peculiarities of the isotherm plots are the relatively cool areas around the compartment periphery. In fact an analysis showed that the maximum wall temperature around the compartment was 98 C, a situation which was hardly believable. A simple analysis of the conduction model reveals a serious defect. The resistance to heat transfer at the wall's interface with the outside air can be compared with the inner wall interface with the interior compartment air using a simple electrical circuit analogy (figure 3.16). Recalling from Chapter II that the conductivity at the inner wall interface was given by the geometric mean between the wall node conductivity and the first node in the compartment, the resistances can be expressed as follows:

OUTER WALL:

$$R_{outer}=\frac{1}{h}$$

where h denotes the heat transfer coefficient.

INNER WALL:

$$R_{inner} = \frac{L}{k_e}$$

where L is the length between the wall node and the first node in the space.

At the start of the problem the value for the outer resistance was found to be 0.012 K/W he inner resistance was found to be about 471 K/W. In this model, heat transfer to the air in the compartment is by conduction only. The very small value for the air conductivity dominates the geometric mean calculation and a large resistance to heat transfer between the surface of the wall and the inner compartment results. Very little heat transfer occurs from the air to the walls. This has a very deleterious effect on the energy balance in the compartment and makes the solutions highly suspect. This basic problem was not realized until run 4 had been completed.

3. Trial 3

Trial 3 was for an open compartment and was run without using the global pressure correction. A 4 ft by 4 ft opening was placed in the west wall. The burn rate was 0.5 lbm/s and the time step was 0.003125 seconds for the entire run (60 seconds). The amount of CPU time required was 181000 CPU seconds. The pressure and heat rate input curves are shown in figure 3.3. The velocity and isotherm plots are given in figures 3.17 to 3.24.

The velocity plots appear to the support the information portrayed by the isotherm plots as the fire progresses. There is a strong mass outflow from the compartment during the entire run. A strong buoyant updraft is shown in the fire's vicinity. Once again the periphery of the compartment is relatively cool. In fact the vent area is much hotter than the adjacent walls.

Upon comparison of the open compartment with the closed compartment it is immediately evident that the results are not what might be expected. The open compartment is several hundred degrees hotter than the closed compartment. The pressure in the open compartment remained at nearly atmospheric pressure while the closed compartment showed almost a linear increase. It is seen that the application of the global pressure correction has major consequences for the solution arrived at. A check on the density fields calculated for the two runs and the use of the perfect gas law serve to illustrate what is occurring. At time 42 seconds representative values of the density field in the two compartments was as follows:

DENSITY (gm/cubic cm)

HEIGHT	OPEN COMPARTMENT	CLOSED COMPARTMENT
1 ft	0.867E-03	0.220E-02
5 ft	0.242E-03	0.810E-03
9 ft	0.144E-03	0.625E-03

The pressure in the closed compartment was 2.545 atmospheres while the open compartment was at 1.0008 atmospheres. Thus the pressure in the closed compartment was 2.54 times greater than that of the open compartment and the density was from 2.5 to 4.3 times greater. Applying the perfect gas law to the open and closed compartments respectively:

$$P_o = \rho_o R T_o$$

$$P_c = \rho_c R T_c$$

Thus,

$$\frac{P_c}{P_o} = \frac{\rho_c}{\rho_o}\frac{T_c}{T_o} \quad ,$$

$$T_c = \frac{2.5}{3.5}T_o = \frac{5}{7}T_o$$

This was arrived at by approximating the pressure and the density of the closed compartment to be 2.5 and 3.5 that of the open compartment respectively. This analysis is far oversimplified but it serves to emphasize the difference between the two solutions and the importance of the pressure field calculation. In general, the air density in the open compartment was less than that of the closed.

The heat input rate in this trial run becomes significantly less than that of the closed compartment as time progresses. The reason is that the radiative heat transfer is much higher and the energy is being continually squandered to the air surrounding the compartment due to the hole in the wall and the conduction model problem mentioned in the trial two run. However, due to the difference in pressure fields the compartment is hotter which serves to increase the radiative transfer. The radiative heat transfer is input back into the heat input loop and the result is the curve of figure 3.3.

Before beginning discussion on trial run 4, one aspect of the open compartment run which needs to be pointed out is the hot spot in the area directly opposite the hole in the wall closest to the fire (figure 3.24). The hot spot does not appear in the closed compartment. In run number 6 the same phenomenon is observed although the flow at the open vent is much different.

4. Trial 4

Trial 4 was conducted to investigate the effect of the absence of the global pressure correction routine on the closed compartment. The total time run was therefore only 20 seconds. The heat input rate and the pressure curves are given in figures. 3.3. The isotherm and velocity plots are given in figures 3.25 to 3.28.

The result of the run was that the pressure field was markedly different from that of the closed compartment with the global correction applied. The pressure rises but not nearly as significantly as in the trial 2 run. The isotherm plot at 20 seconds is similar but the magnitude of the temperature field is higher. From this run it was decided to attempt to incorporate a global pressure correction for the open compartment.

5. Trial 5

The intent of trial run 5 was to experiment with the conduction model so that the resistance between the wall and the interior compartment was of about of the same magnitude as that between the wall and the exterior of the compartment. The geometric mean conductivity for the inner wall was increased by a factor of 1000.

For this run a time step of 0.00935 seconds was used for the first 20 seconds and 0.0125 seconds was used for the rest of the run. The heat input and pressure curves are as given in figure 3.4. The total CPU time required was 44250 seconds.

Since the intent of this run was to investigate the results of an alteration to the conduction model on the heat transfer to the wall, a minimum of plots are provided (figures 3.29 to 3.31). The maximum temperatures of the walls have now increased dramatically. At 60 seconds the wall closest to the fire has reached over 1000 C in spots. The isotherms indicate that the area near the walls are several hundred degrees hotter than that of trial 2. A close inspection shows that the periphery of the compartment is now mostly blue rather than black indicating a marked temperature rise. However, it still appears that the effect of radiative heat transfer is not being registered in the walls. The bottom wall, for instance, is hardly above ambient (maximum 70 degrees) at 60 seconds. It would be expected that there should be some indication of heat being transferred to the air from the floor but such is not the case. The heat transfer is apparently mostly from that convected from the fire to the vicinity of the walls.

6. Trial 6

The final trial was for an open compartment with the global pressure correction applied to the interior and with the geometric mean conductivity of the inner walls increased by a factor of 1000 as in trial 5. The hole was of the same size as that used in trial 3. A time step of 0.00625 seconds was used for the entire run. The total CPU time required was 79285 seconds. The heat input rate and pressure curves are given in figure 3.4. The velocity and isotherm plots are given in figures 3.32 to 3.36.

The result of the application of the global pressure correction by incorporating the mass transfer from the compartment is seen in the pressure curve. The pressure in the open compartment is higher than that of the closed compartment at 60 seconds. The velocity diagrams show a slight inflow at the bottom of the opening in the wall and no flow at the top portion of the opening. It is difficult to distinguish any flow through the opening. The isotherm plots however, indicate that the temperature field is slightly cooler than for the open compartment. The fact that there is an opening in the wall opposite the fire cannot be distinguished from the isotherm plots. However, the hot spot in the wall adjacent to the fire has reappeared and is similar to that alluded to in trial 3.

The boundary conditions implemented for this run were identical to that of the run in trial 3. The global pressure correction was applied to only those control volumes in the compartment. It was not applied to the control volumes immediately outside of the compartment interior. The pressure calculated for the nodes outside of the boundary was 1 atmosphere (as intended) after 60 seconds had elapsed. The pressure of the nodes immediately inside the boundary was over 3 atmospheres. Surely some outward flow should have developed. This aspect of the analysis is still being investigated.

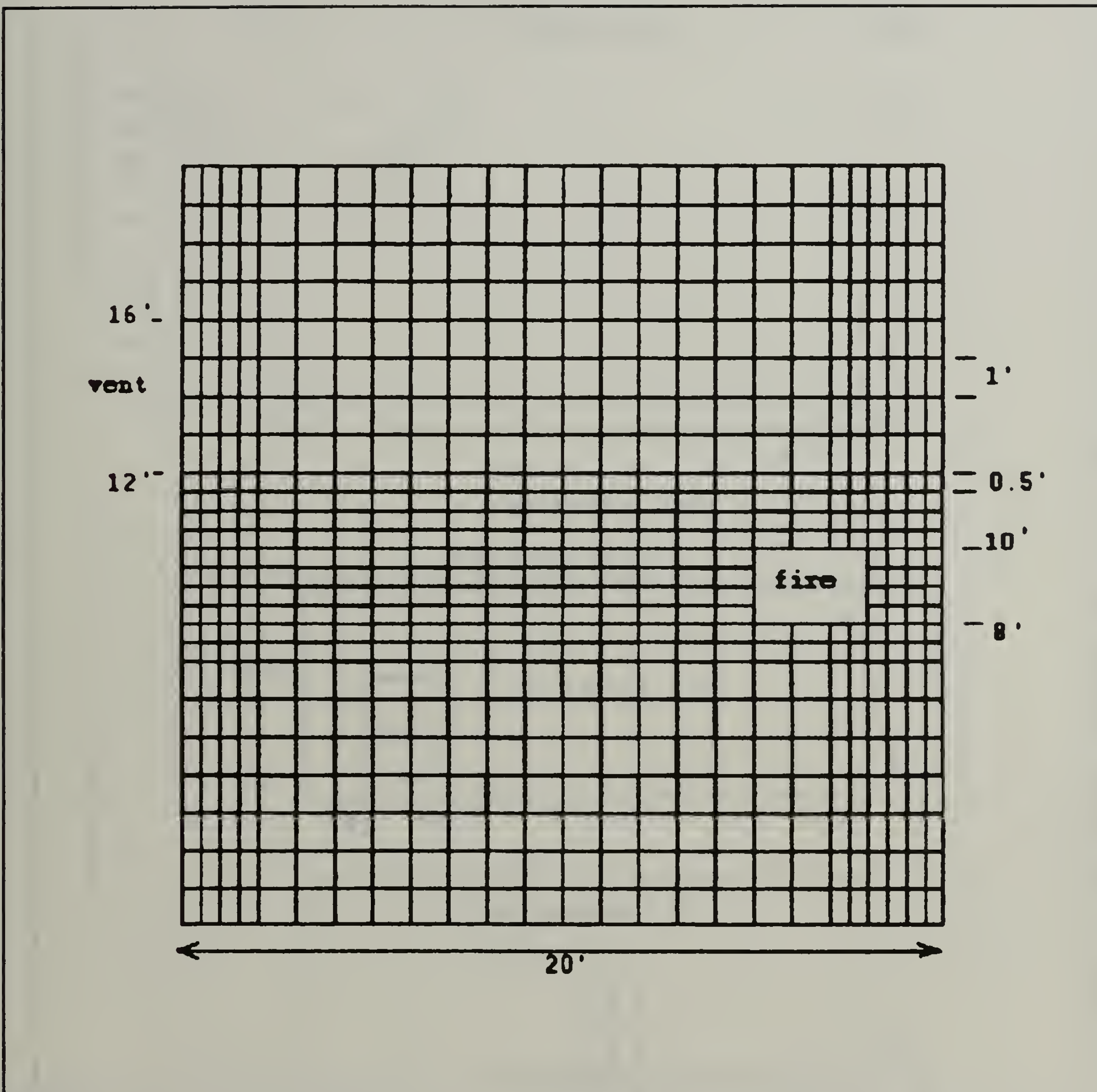


Figure 3.1 Grid used

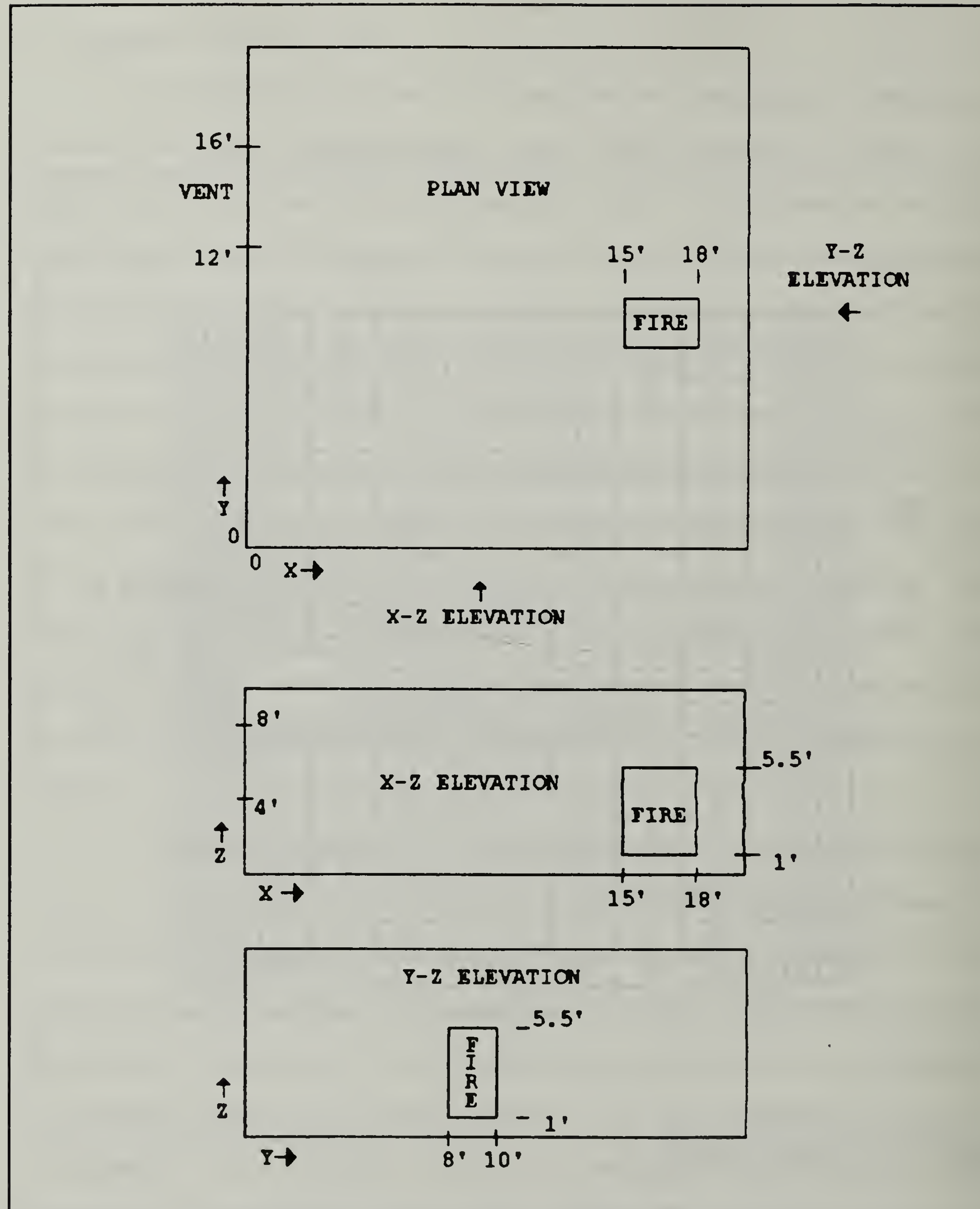


Figure 3.2 Graphics plots orientation

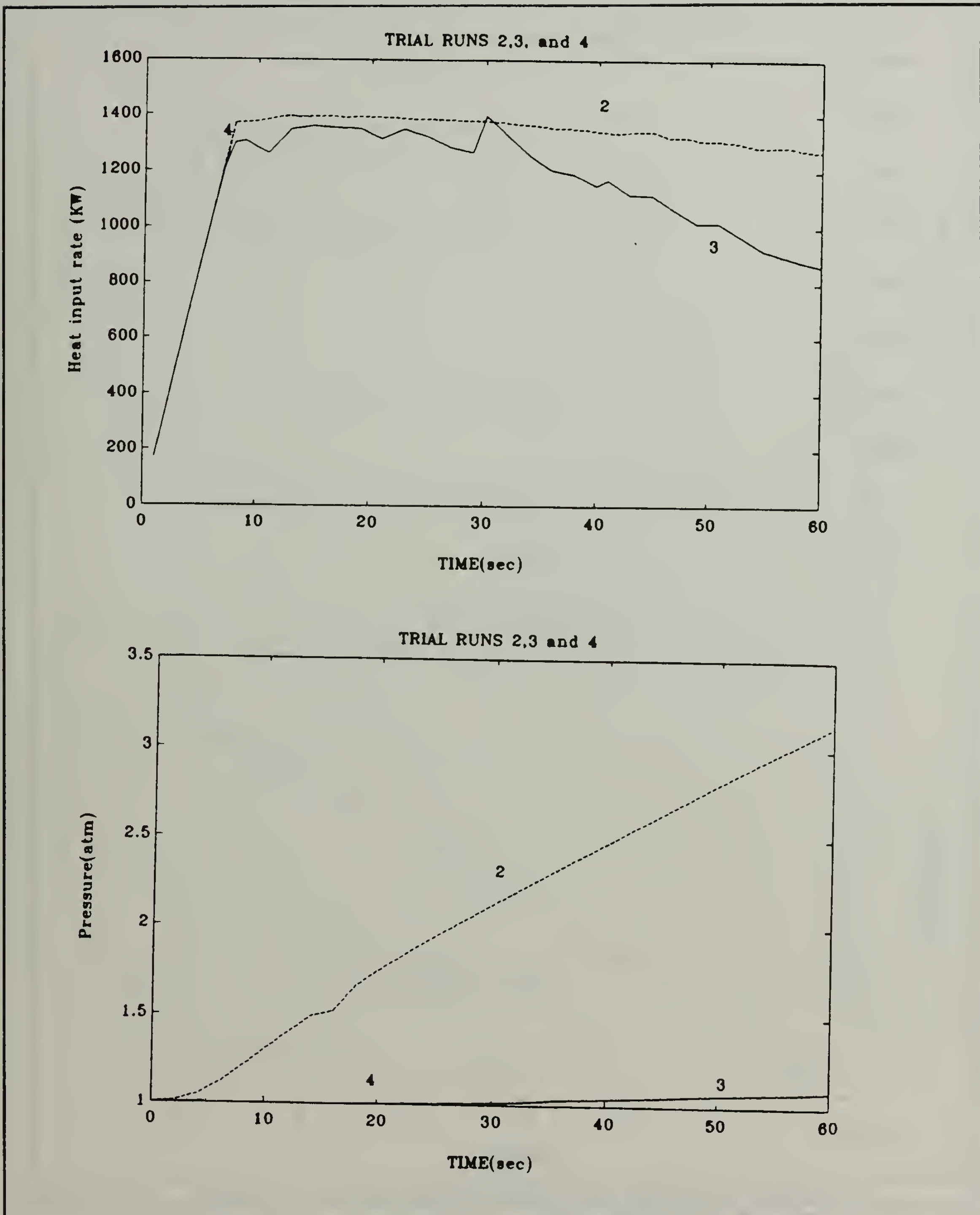


Figure 3.3 Trials 2,3,4 Heat Input/Pressure Curves

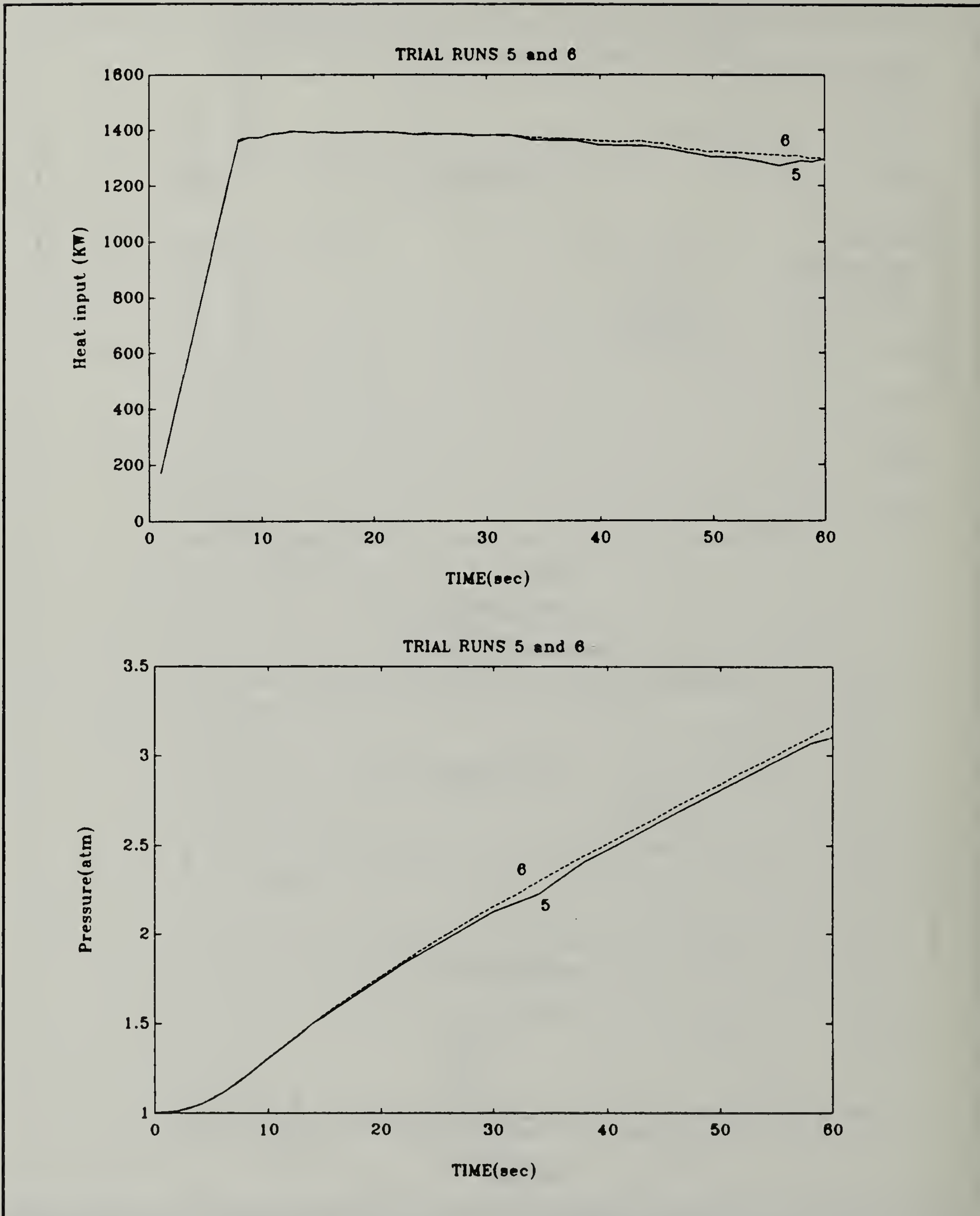


Figure 3.4 Trials 5,6 Heat Input/Pressure Curves

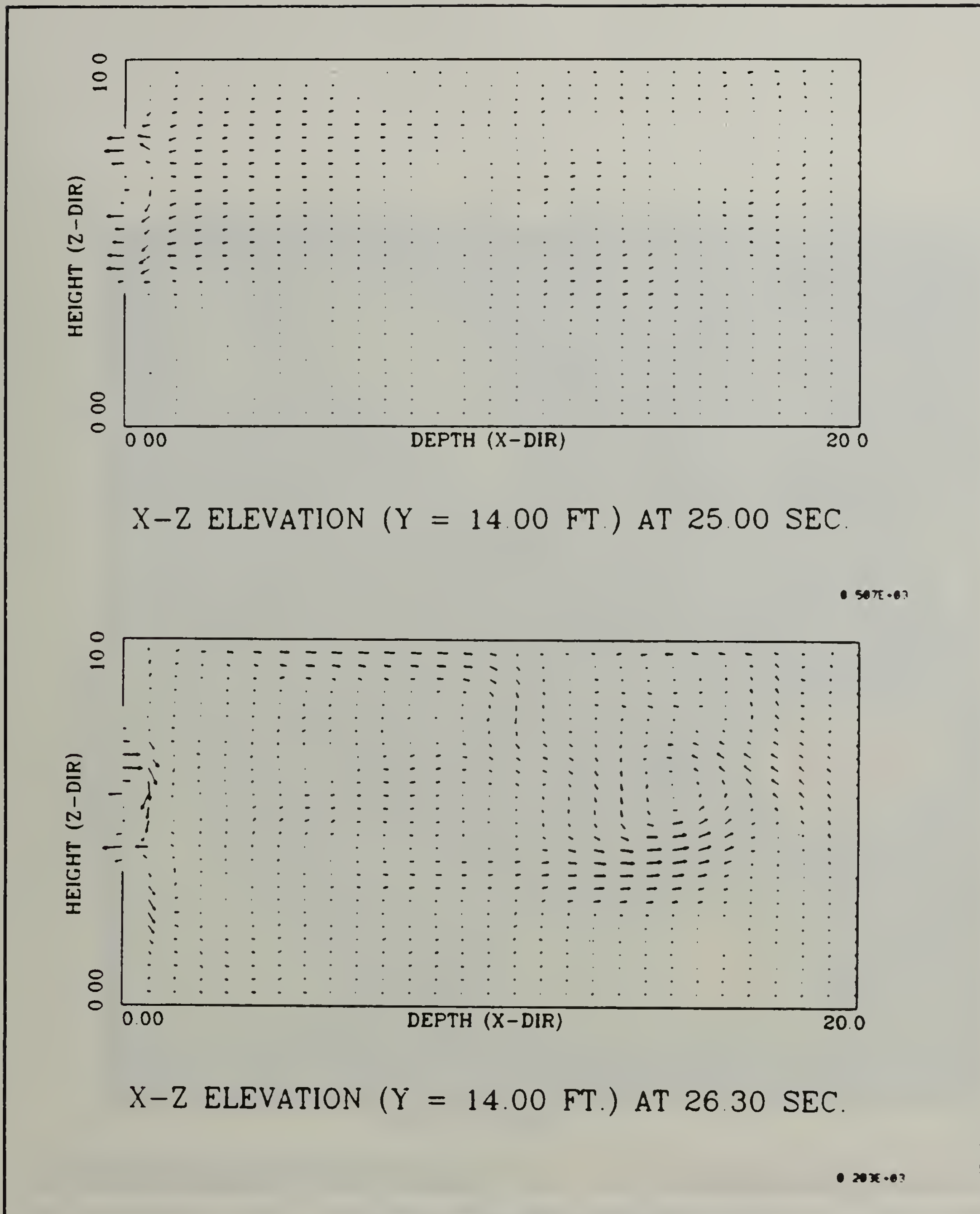


Figure 3.5 Trial 1 Velocity Profile X-Z Elev 25,26.3s

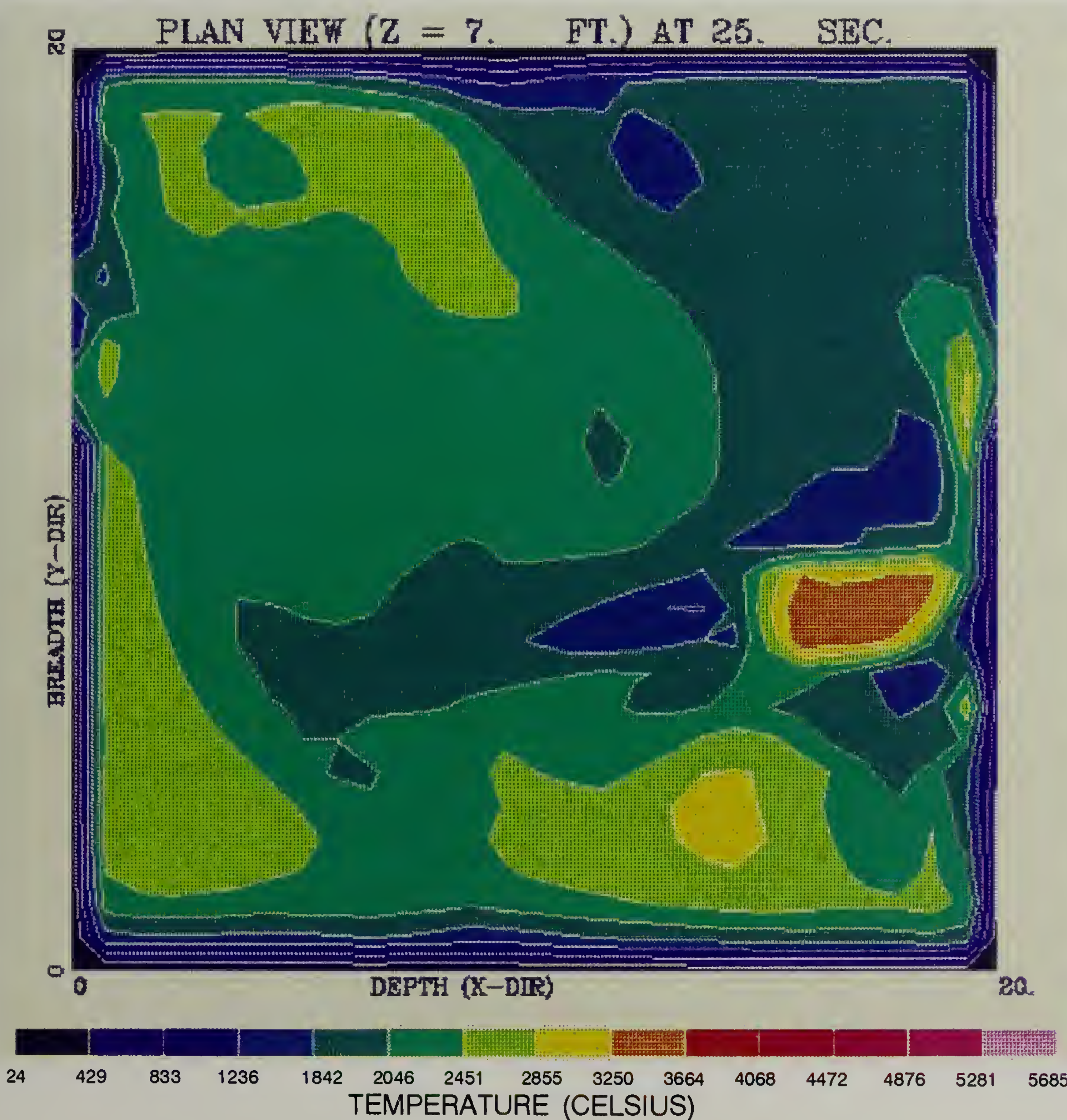


Figure 3.6 Trial 1 Isotherm Profile Plan View at 25s

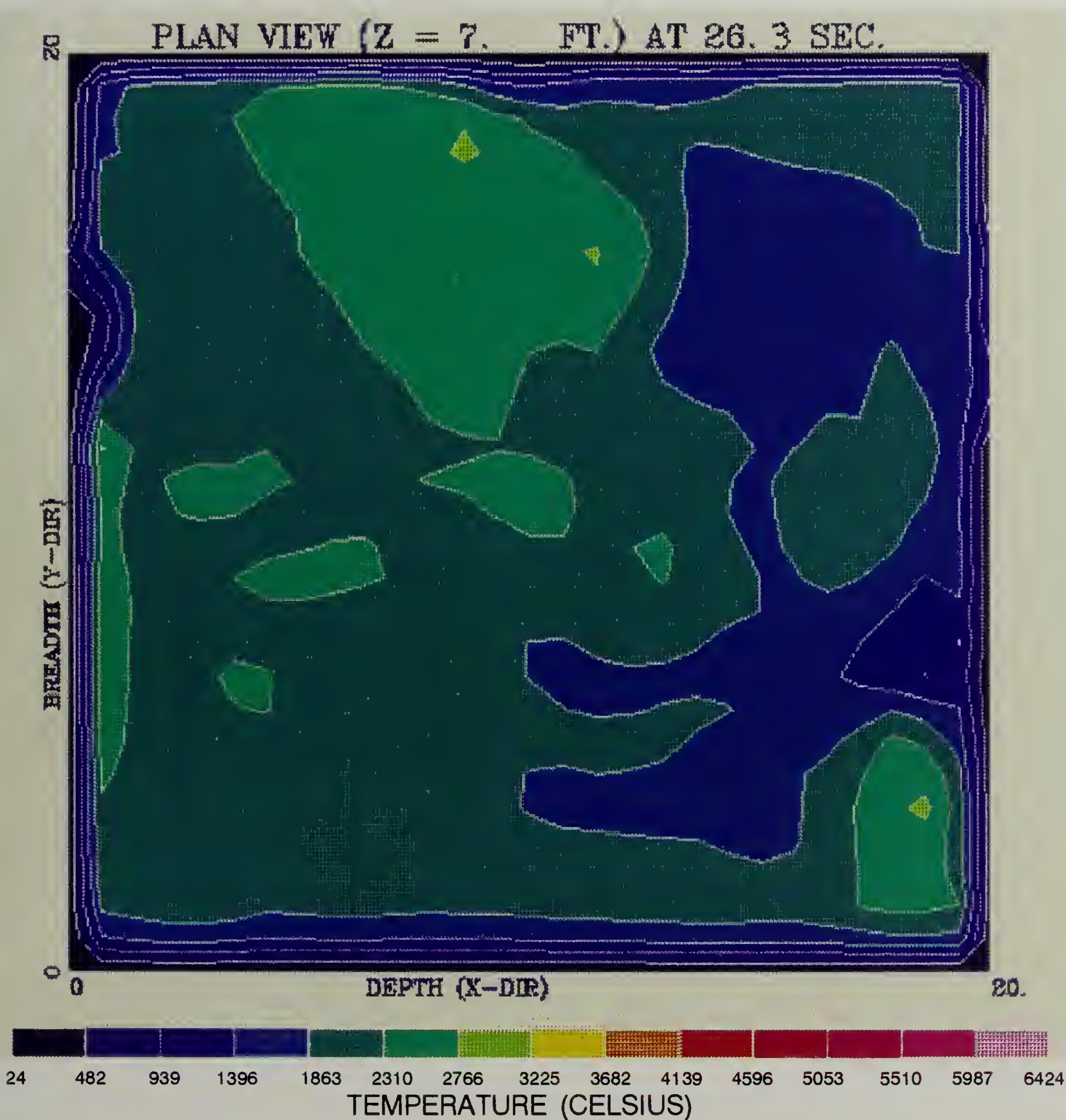


Figure 3.7 Trial 1 Isotherm Profile Plan View at 26.3s

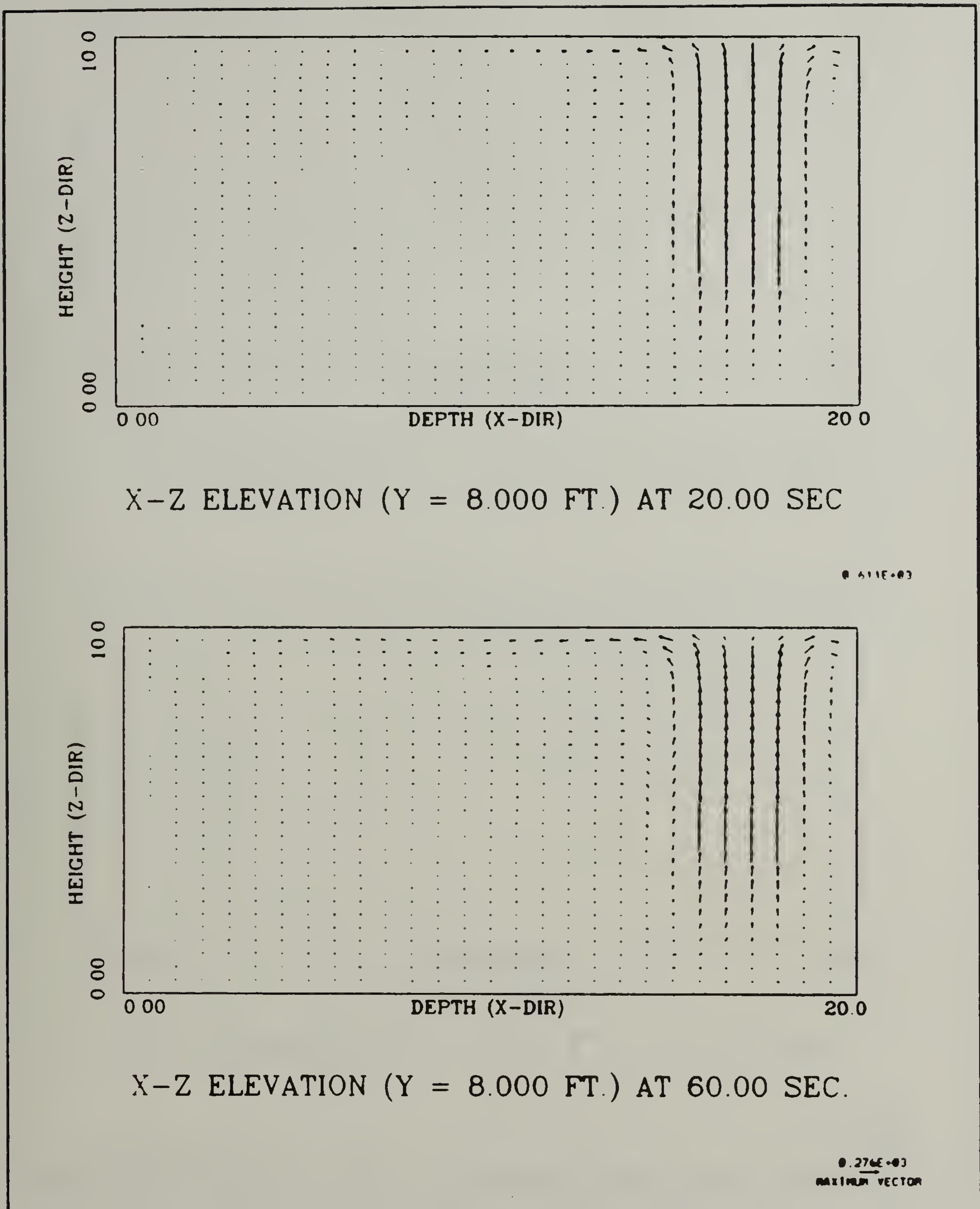


Figure 3.8 Trial 2 Velocity Profile X-Z Elev at 20,60s

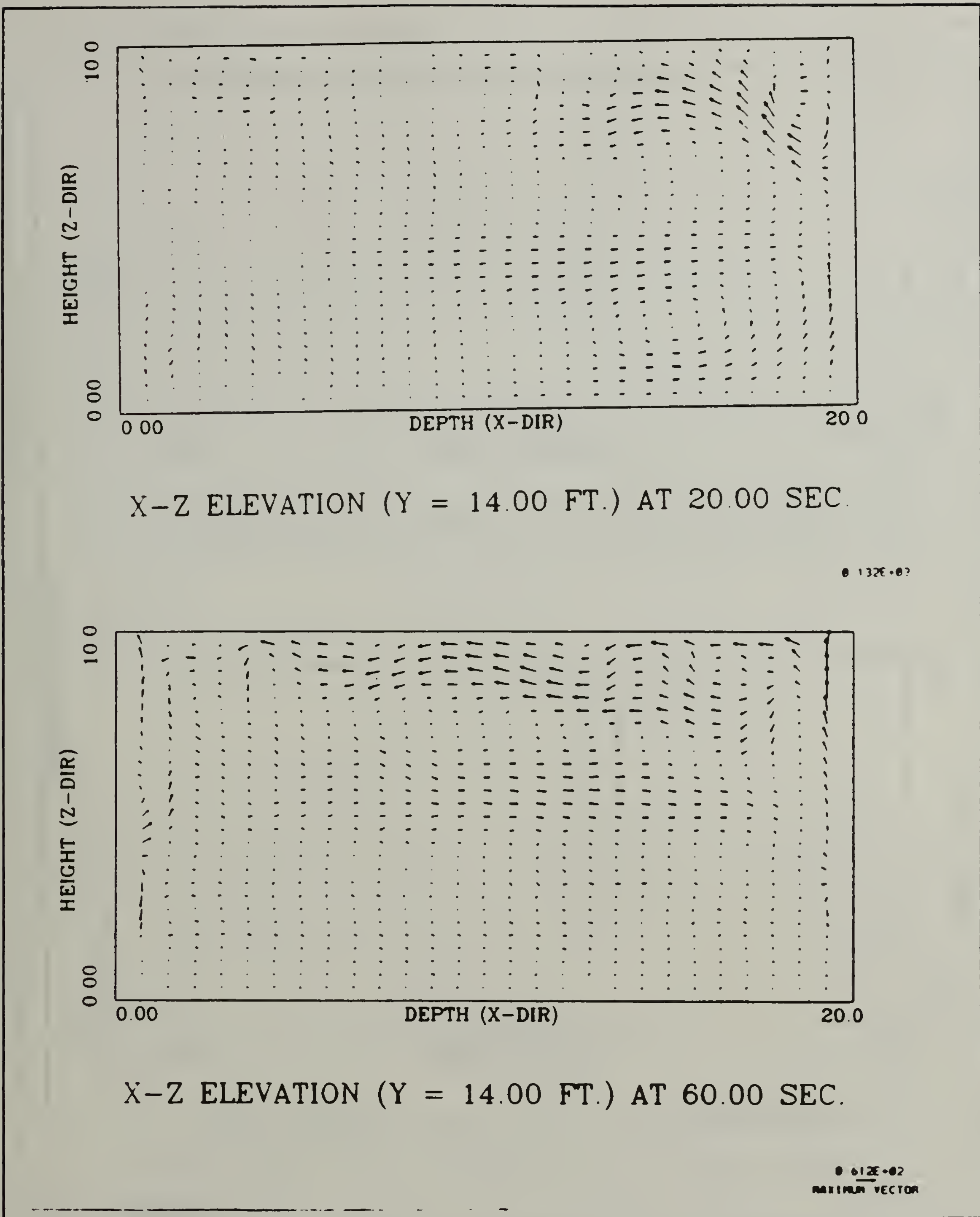


Figure 3.9 Trial 2 Velocity Profile Elev at 20,60s

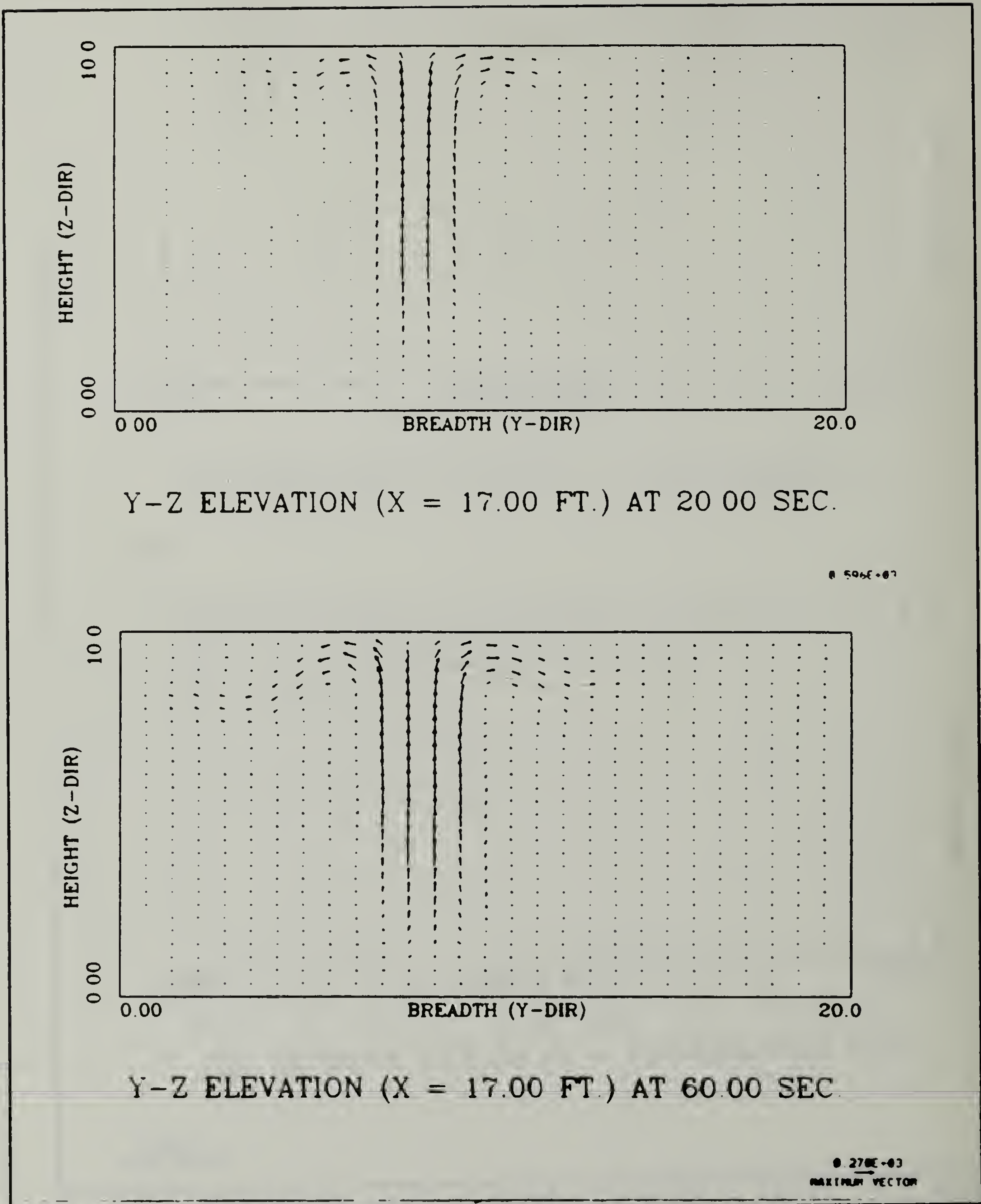


Figure 3.10 Trial 2 Velocity Profile Elev at 20,60s

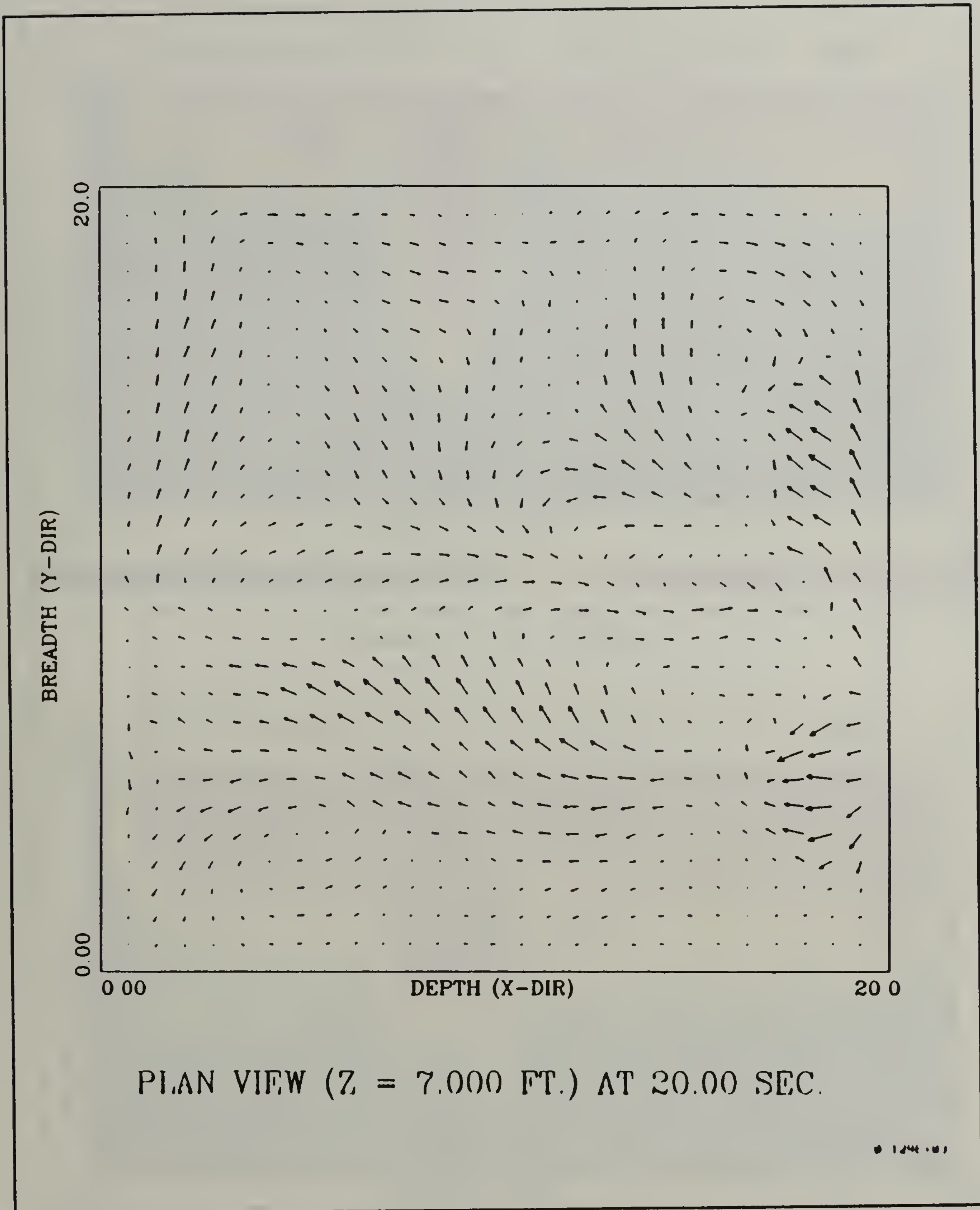


Figure 3.11 Trial 2 Velocity Profile Plan View at 20s

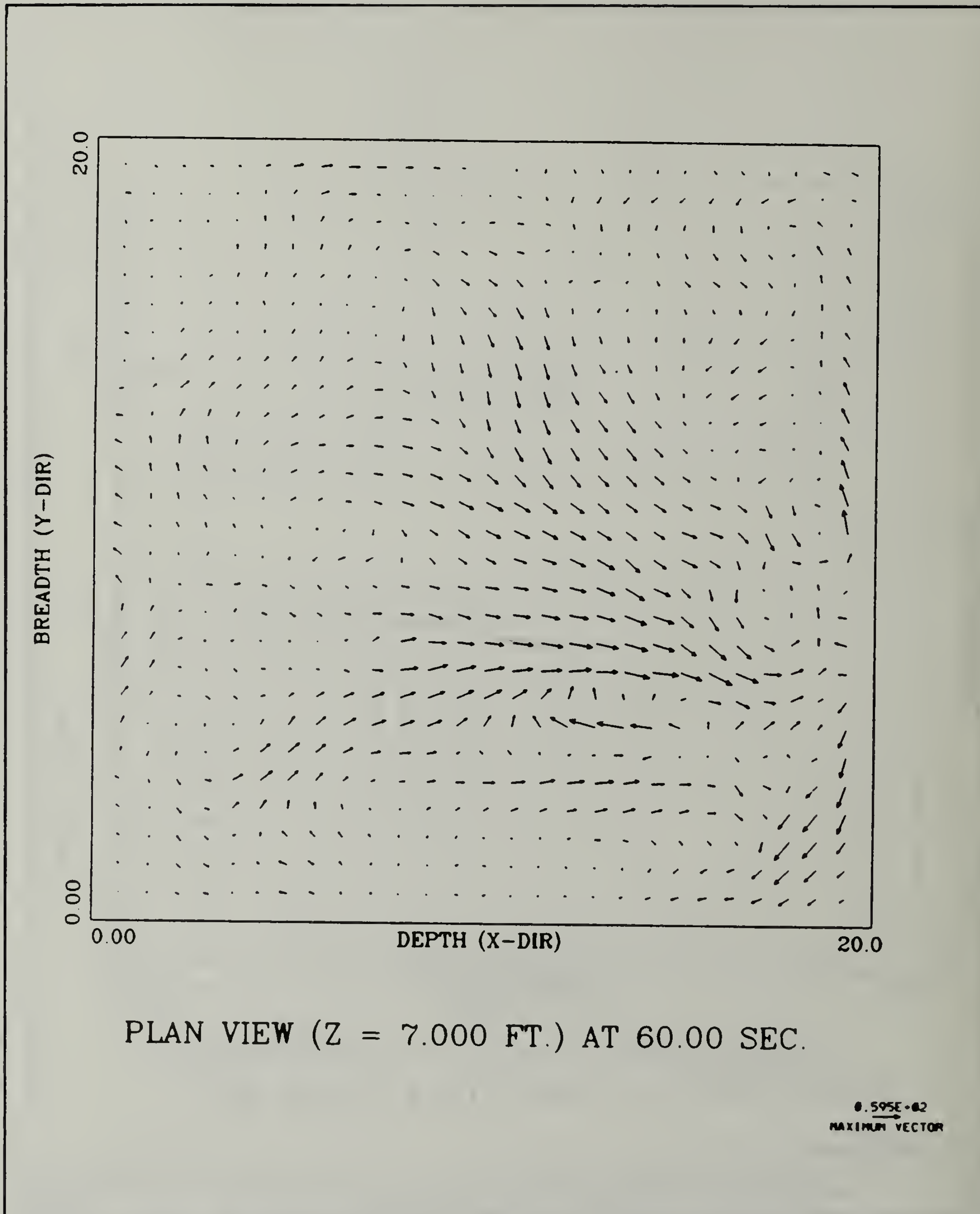
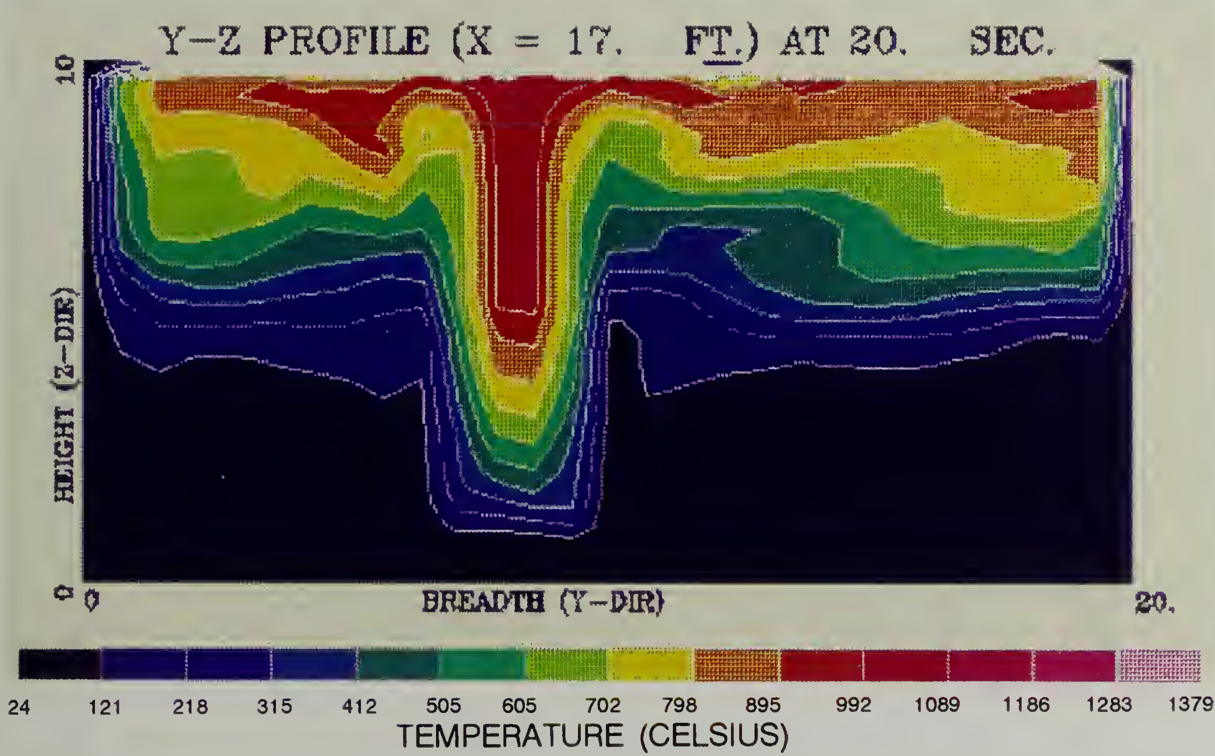


Figure 3.12 Trial 2 Velocity Profile Plan View at 60s



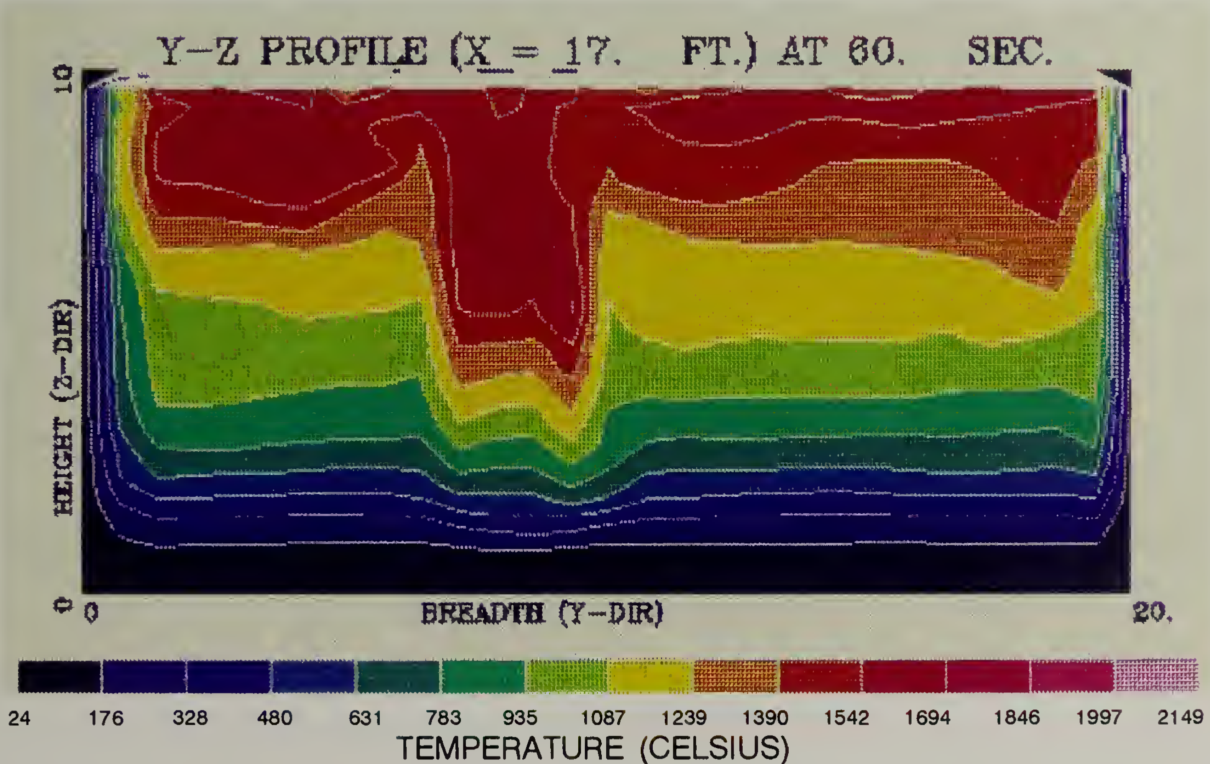


Figure 3.13 Trial 2 Isotherm Profile Elev at 20 and 60s

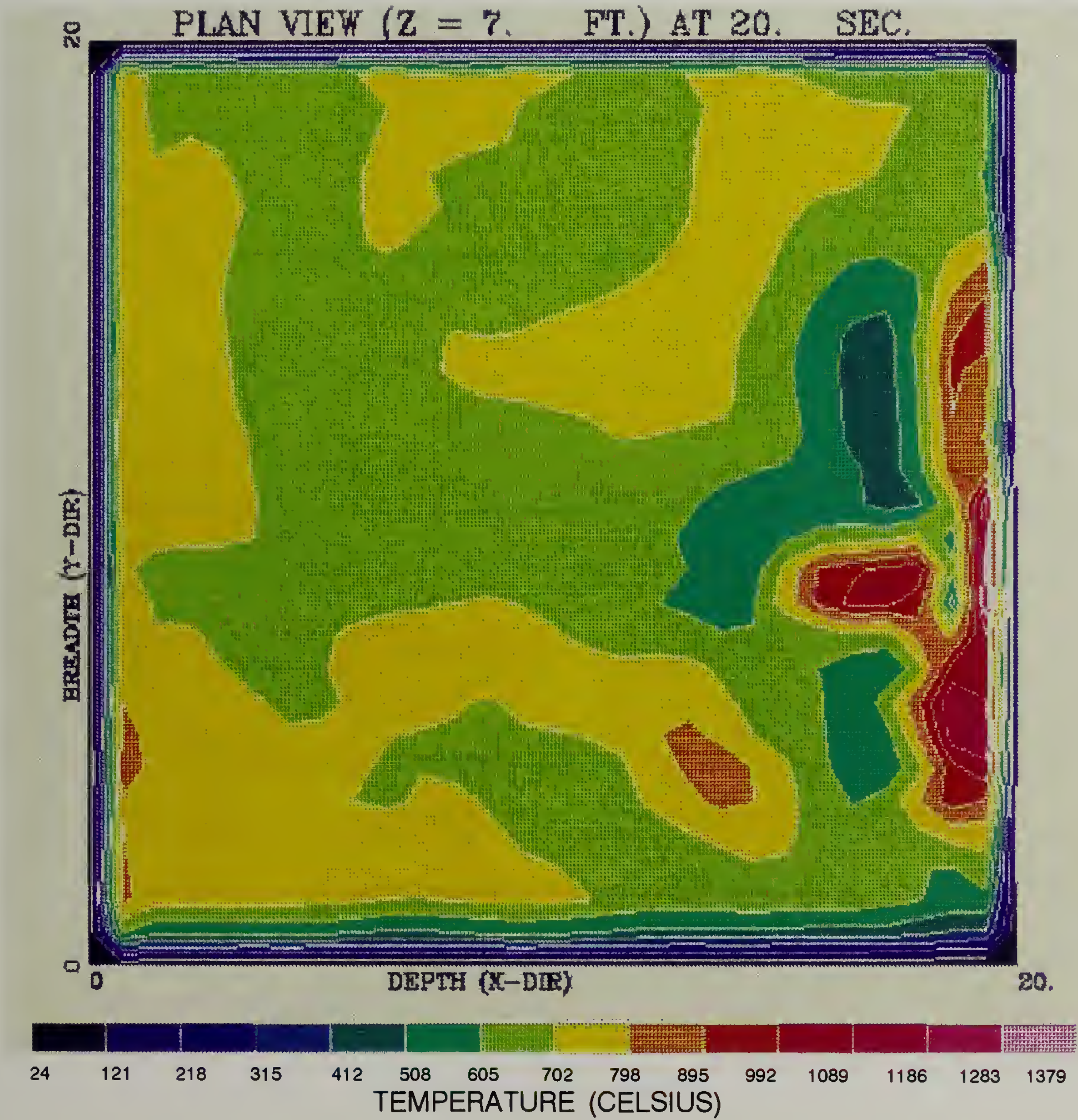


Figure 3.14 Trial 2 Isotherm Profile Plan View at 20s

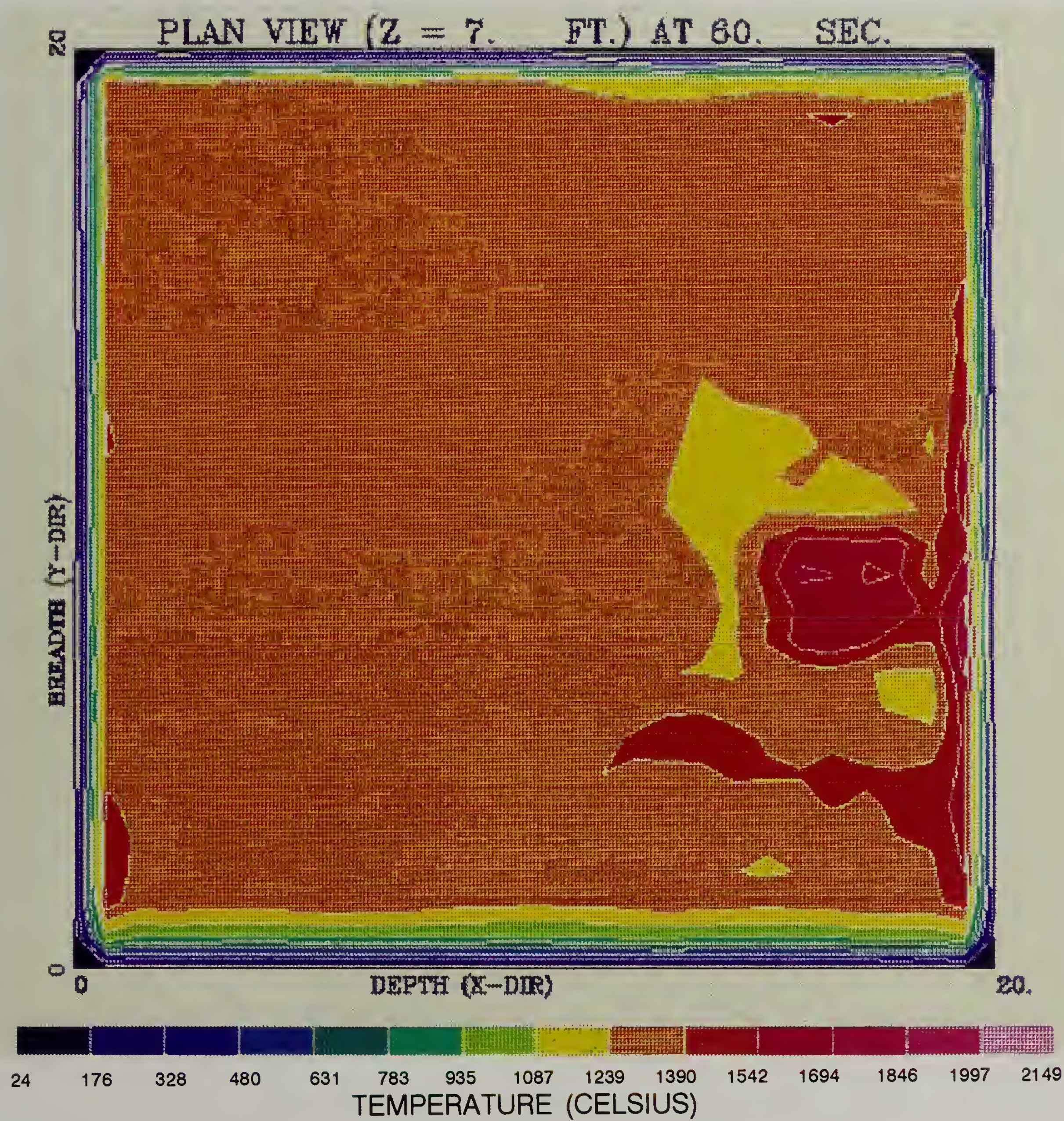


Figure 3.15 Trial 2 Isotherm Profile Plan View at 60s

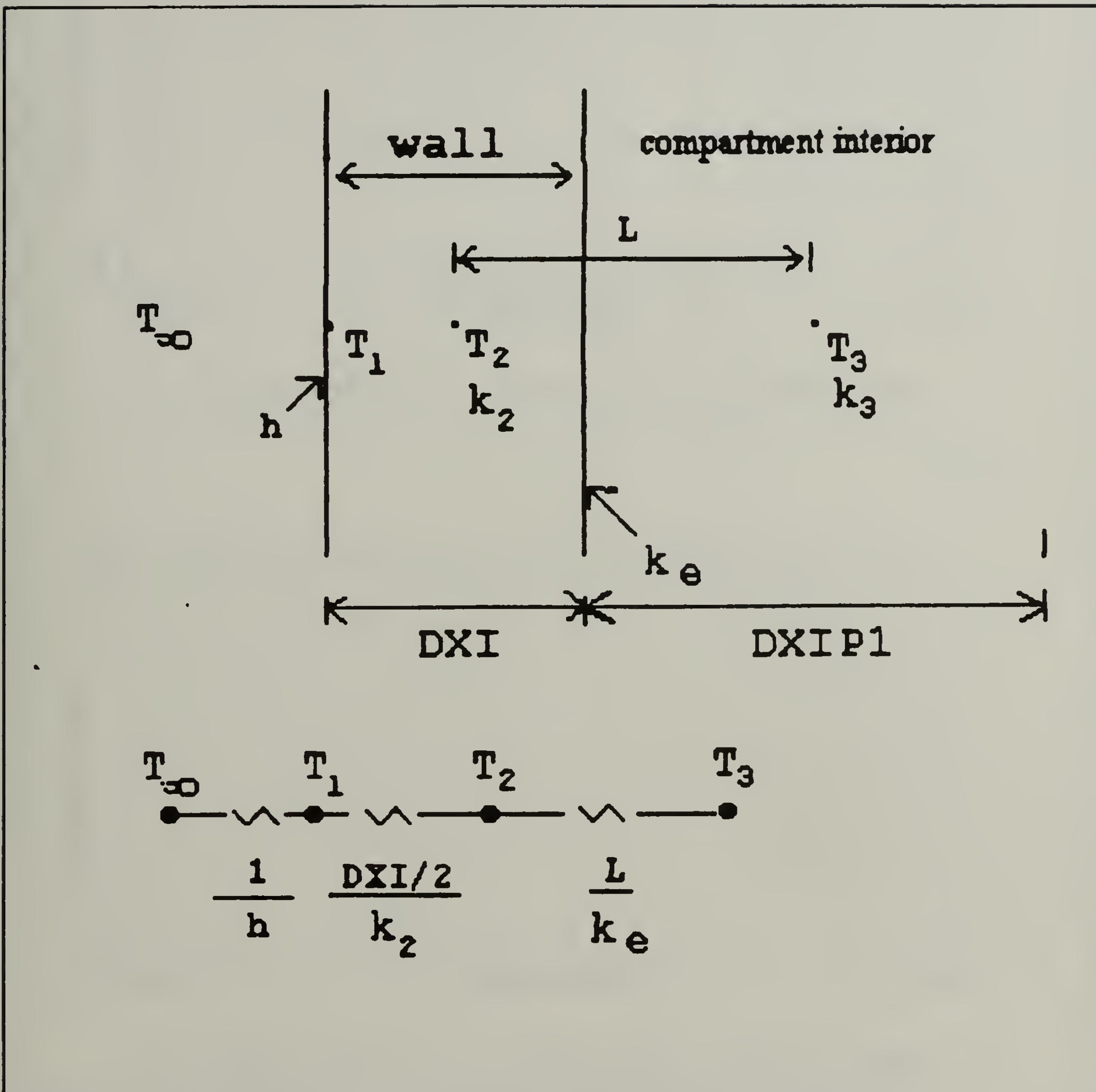


Figure 3.16 Conduction Model Resistance Diagram

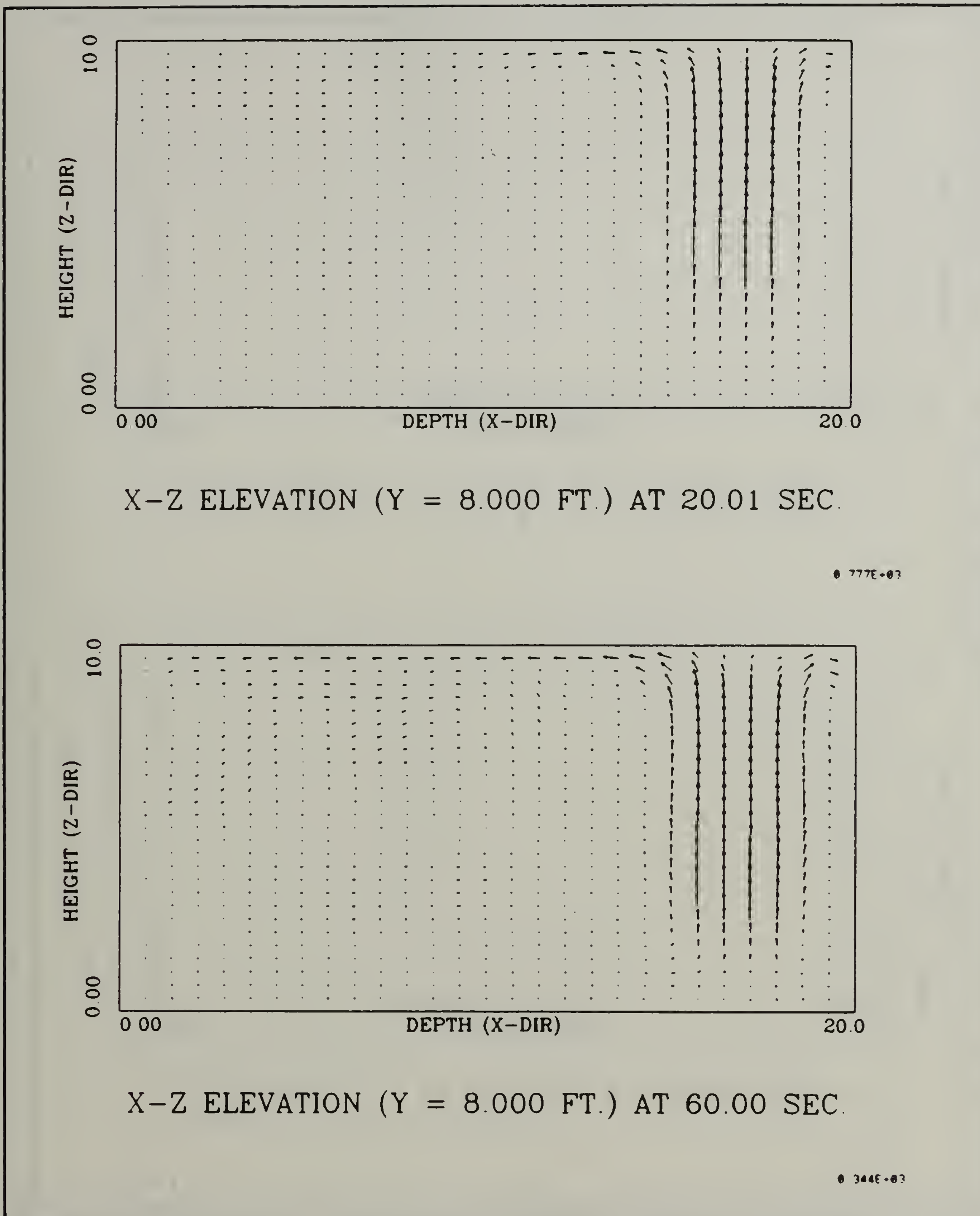


Figure 3.17 Trial 3 Velocity Profile Elev at 20,60s

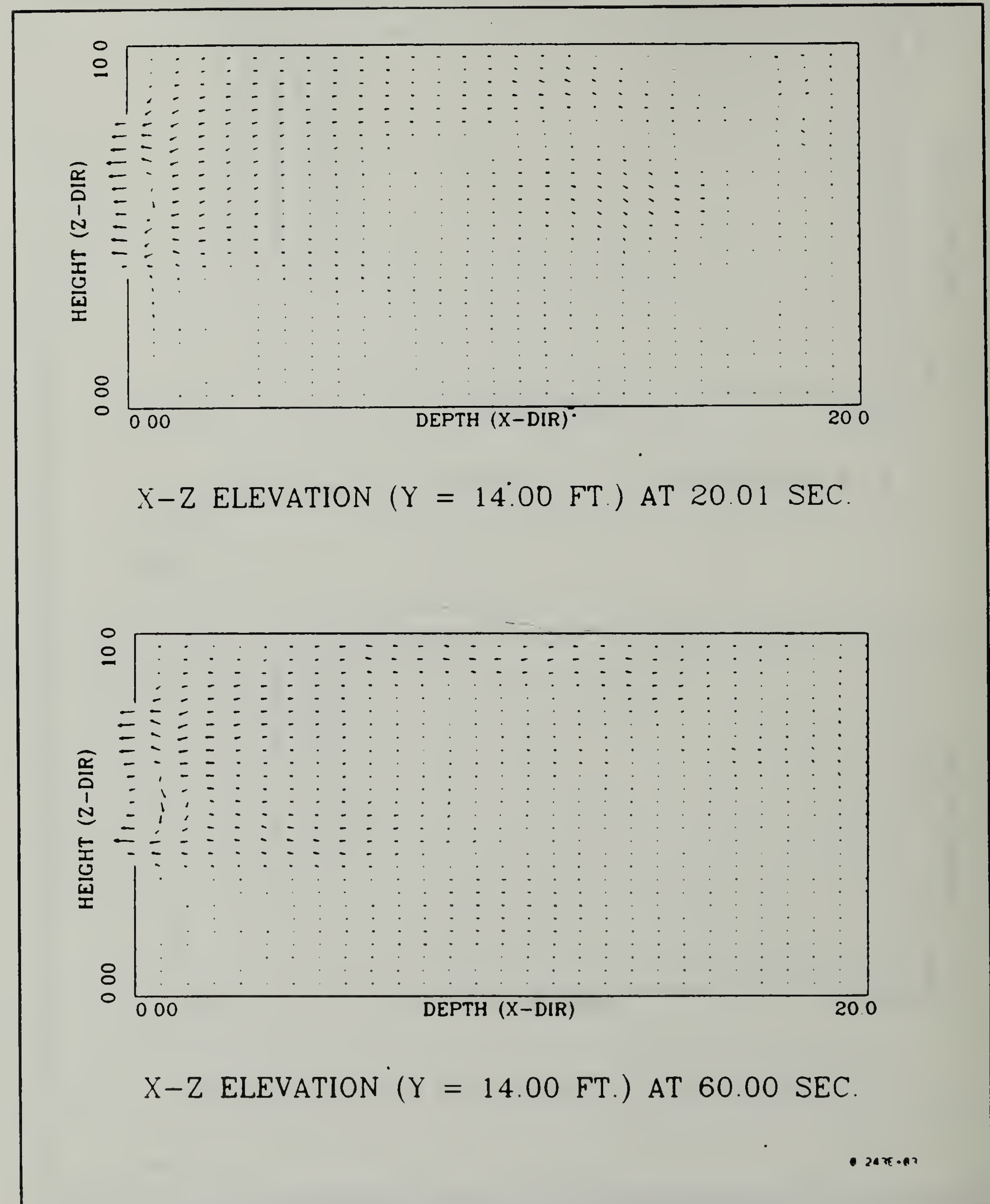
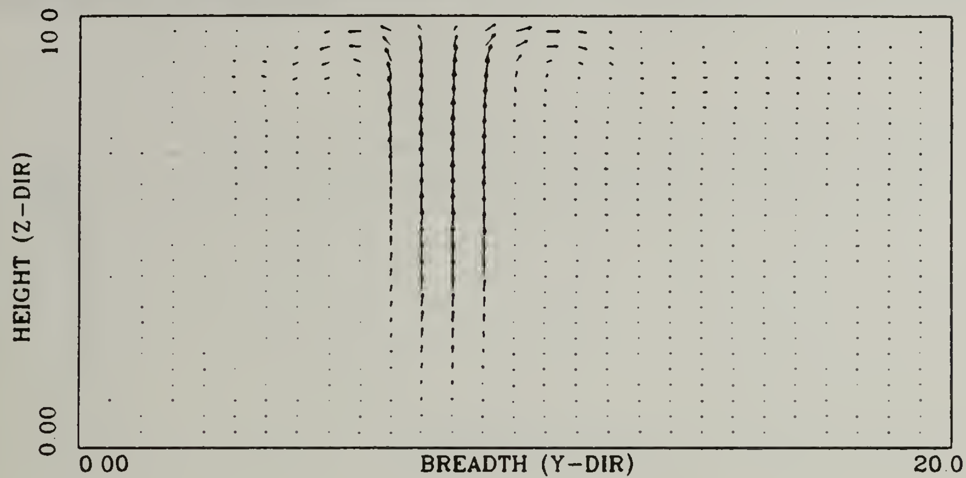
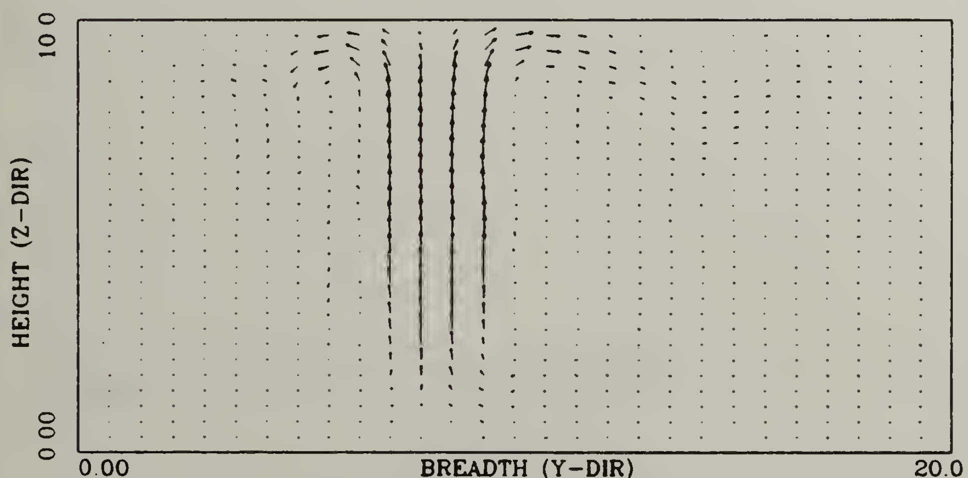


Figure 3.18 Trial 3 Velocity Profile Elev at 20,60s



Y-Z ELEVATION (X = 17.00 FT.) AT 20.01 SEC.



Y-Z ELEVATION (X = 17.00 FT.) AT 60.00 SEC.

Figure 3.19 Trial 3 Velocity Profile Elev at 20,60s

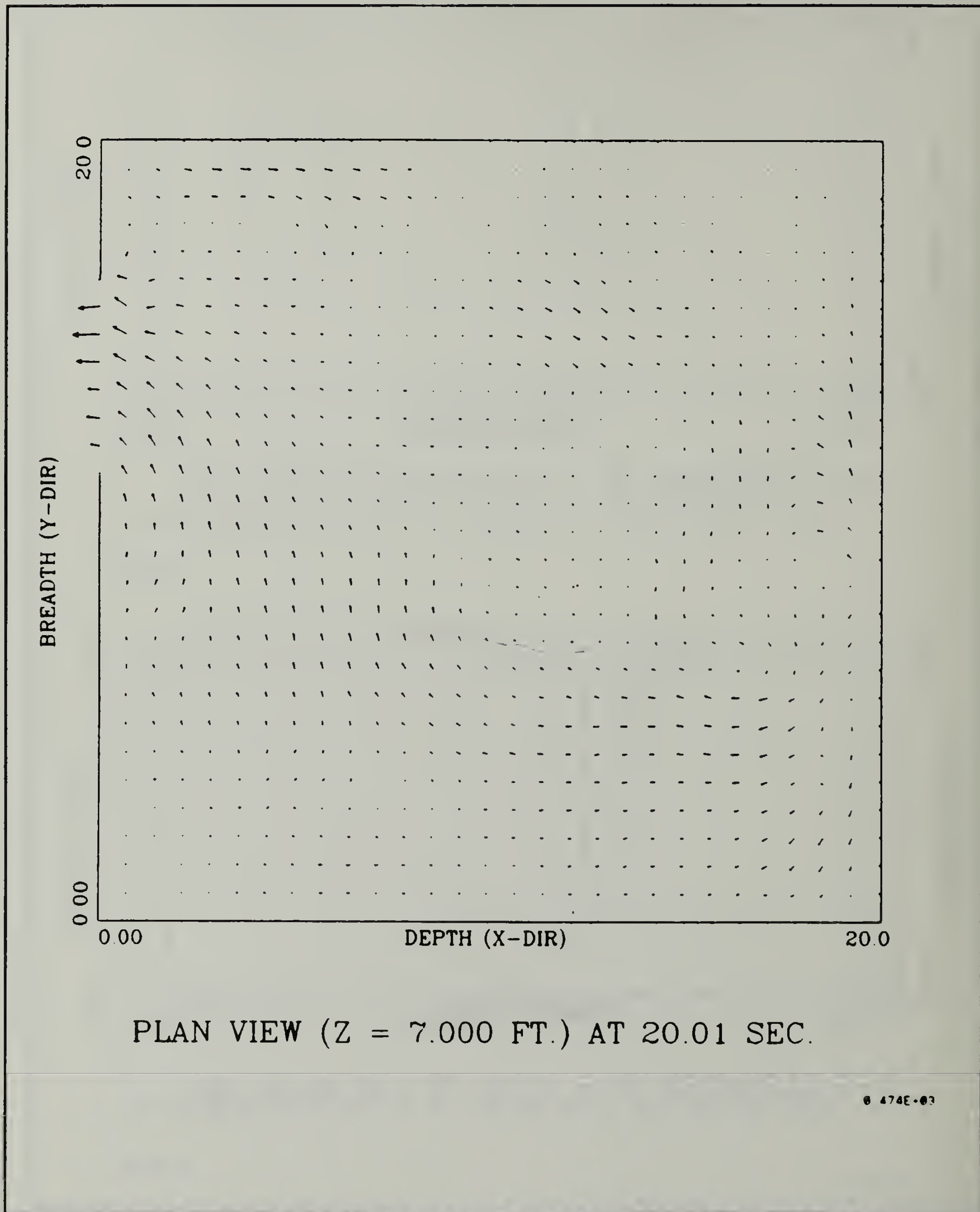


Figure 3.20 Trial 3 Velocity Profile Plan View at 20s

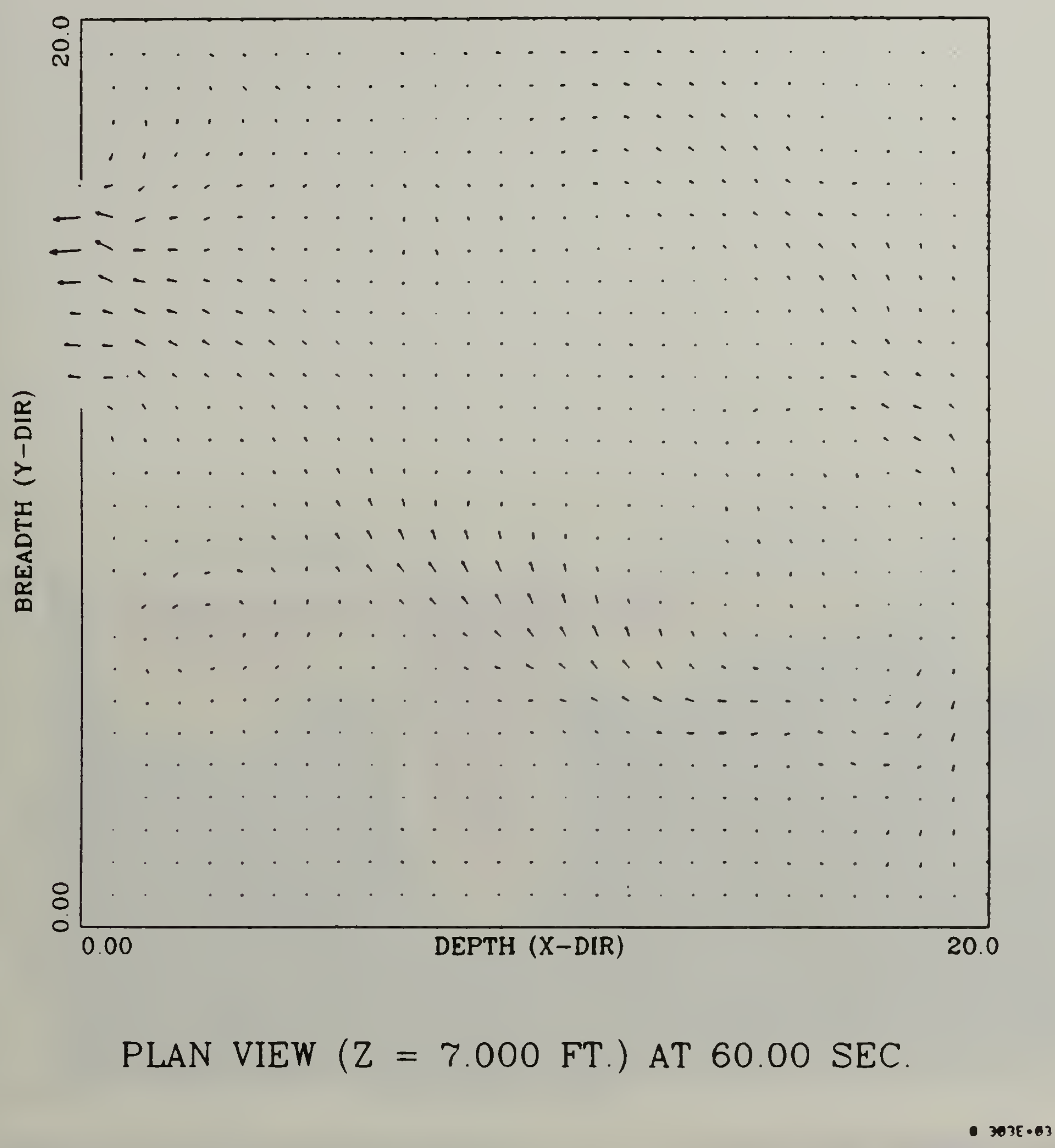


Figure 3.21 Trial 3 Velocity Profile Plan View at 60s

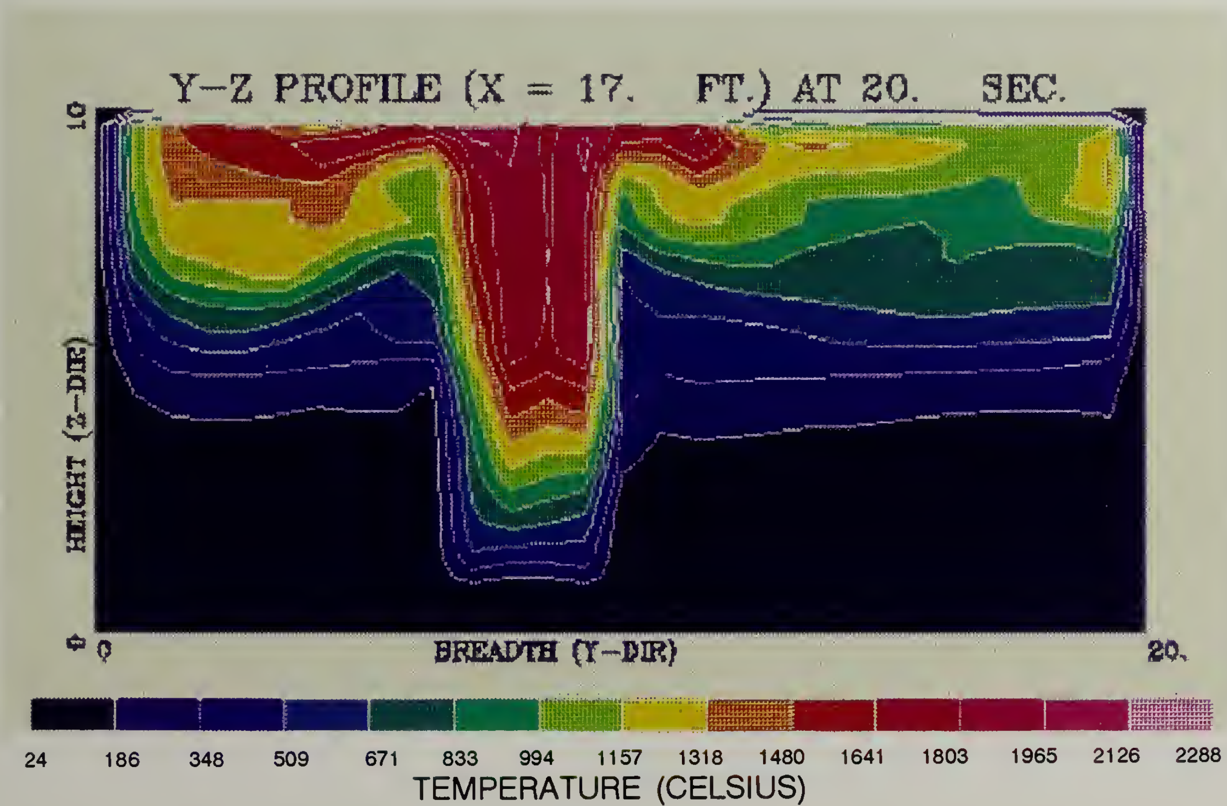


Figure 3.22 Trial 3 Isotherm Profile Elev at 20s

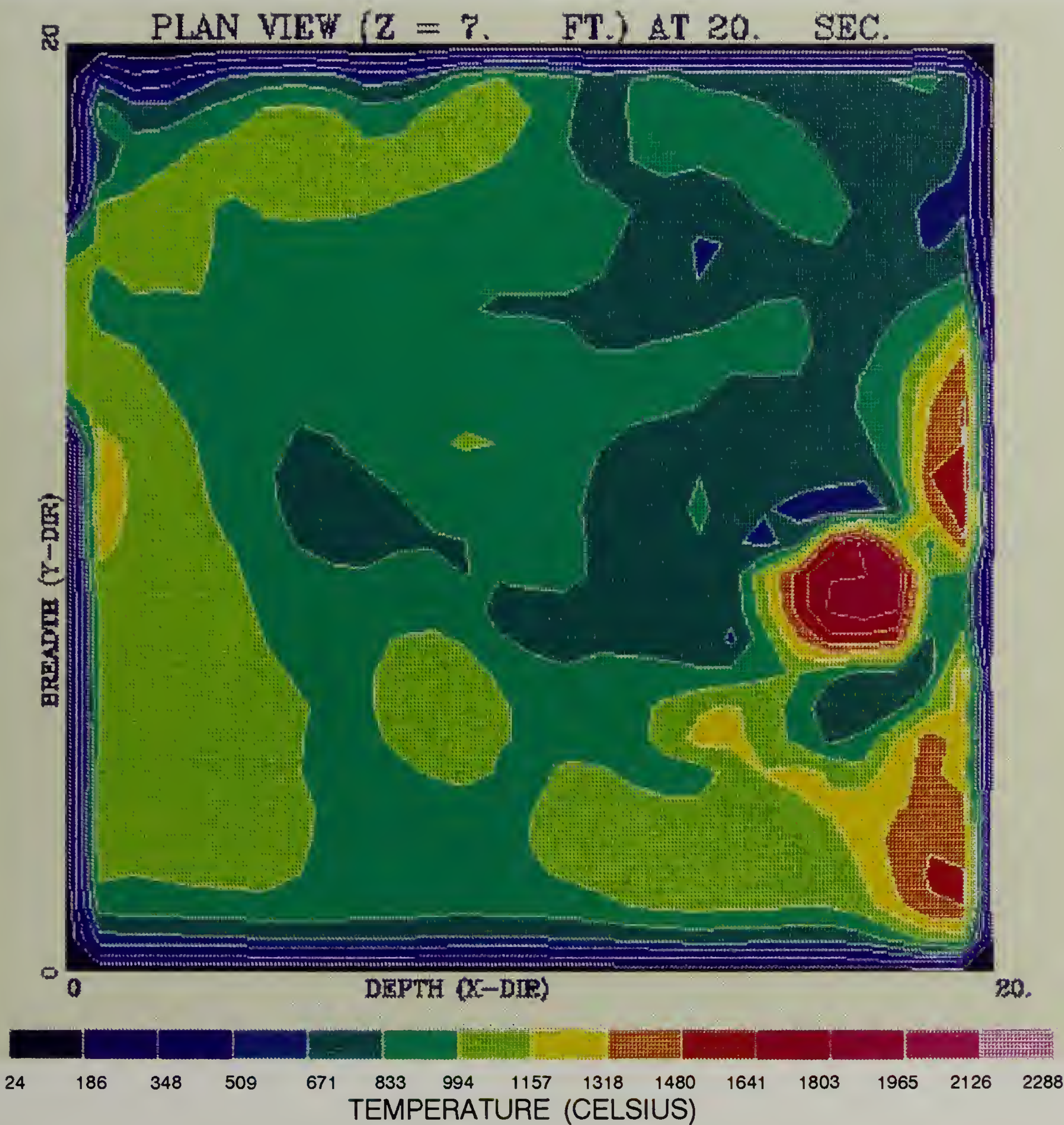


Figure 3.23 Trial 3 Isotherm Profile Plan View at 20s

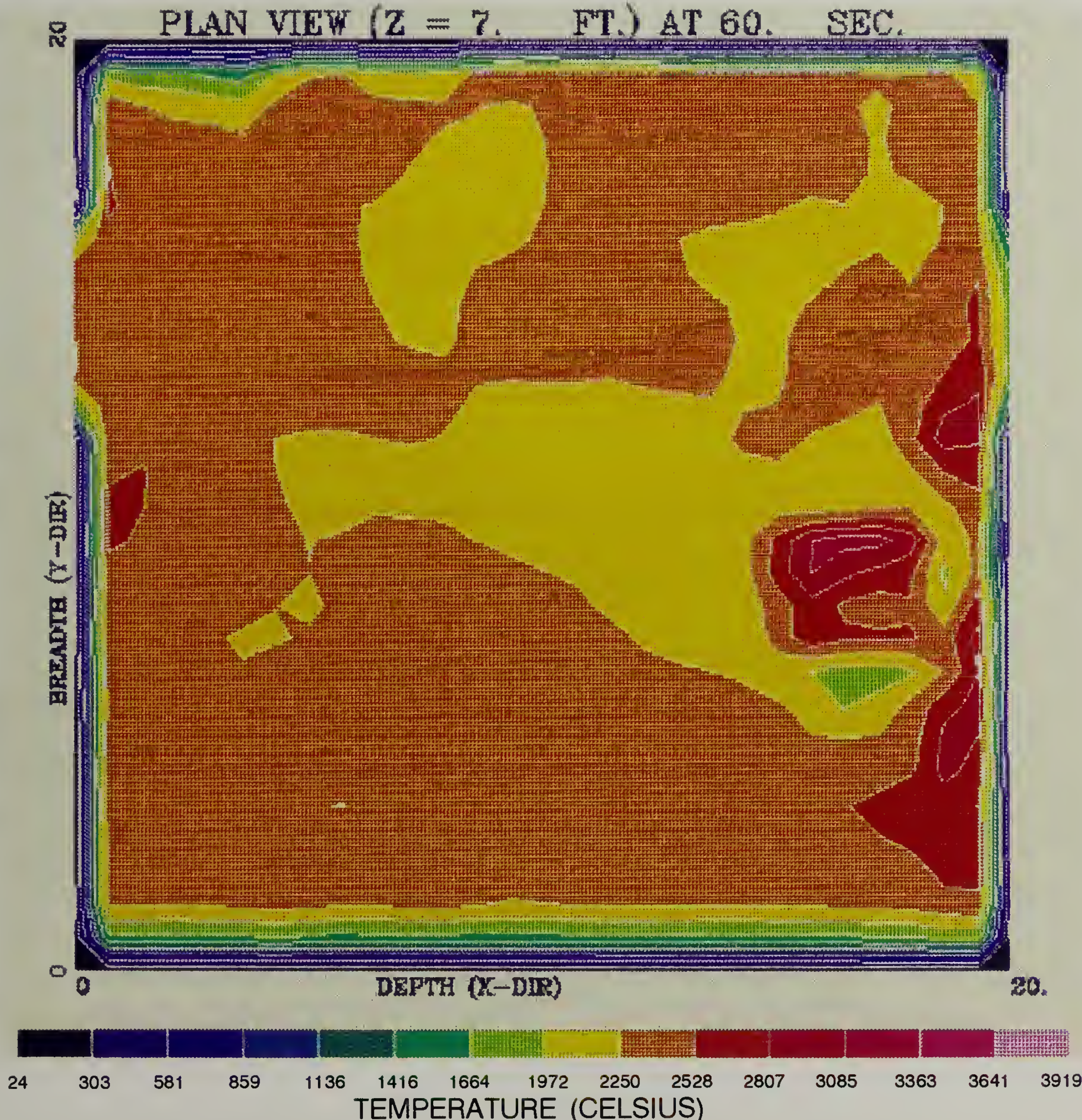


Figure 3.24 Trial 3 Isotherm Profile Plan View at 60s

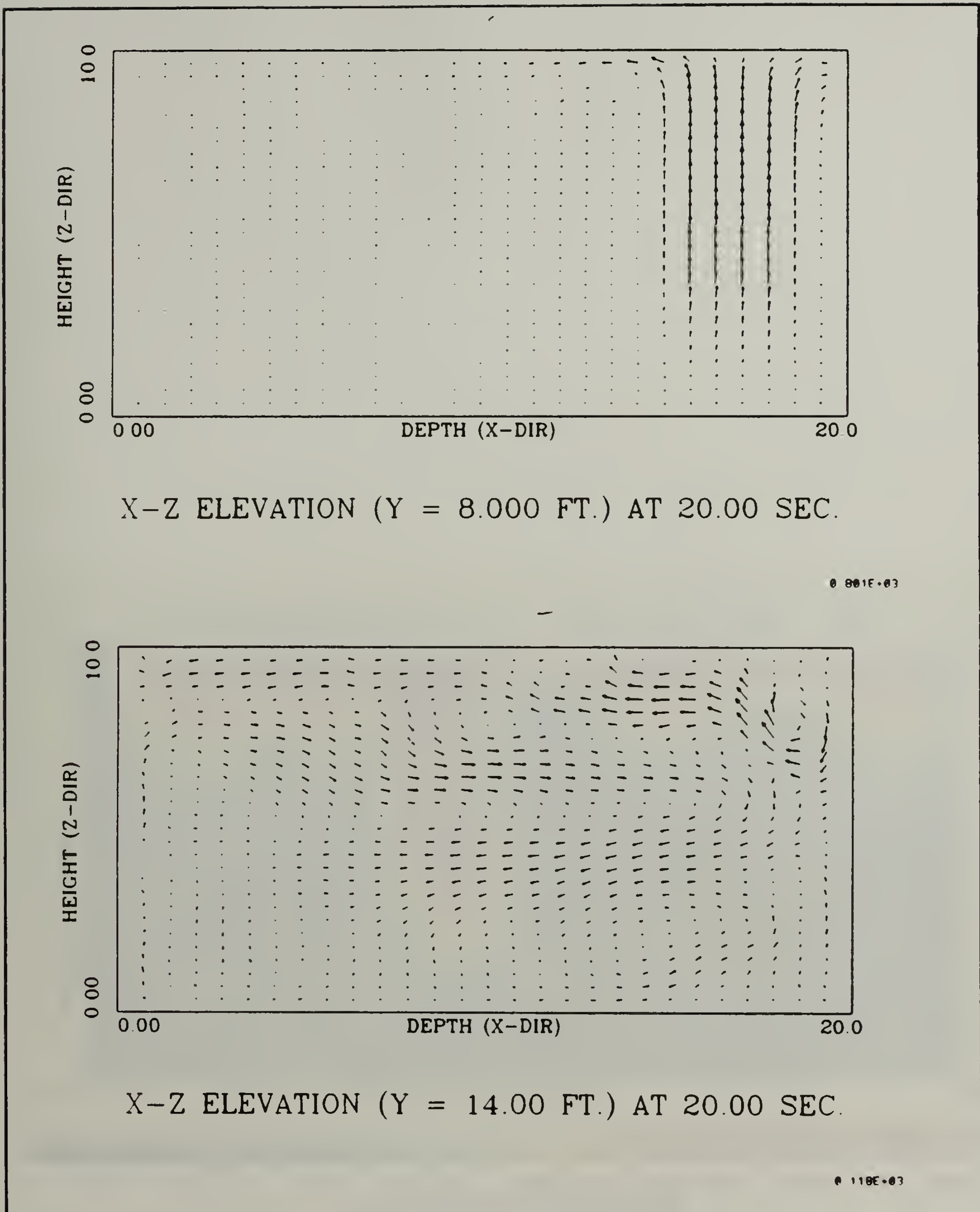


Figure 3.25 Trial 4 Velocity Profile Elev at 20s

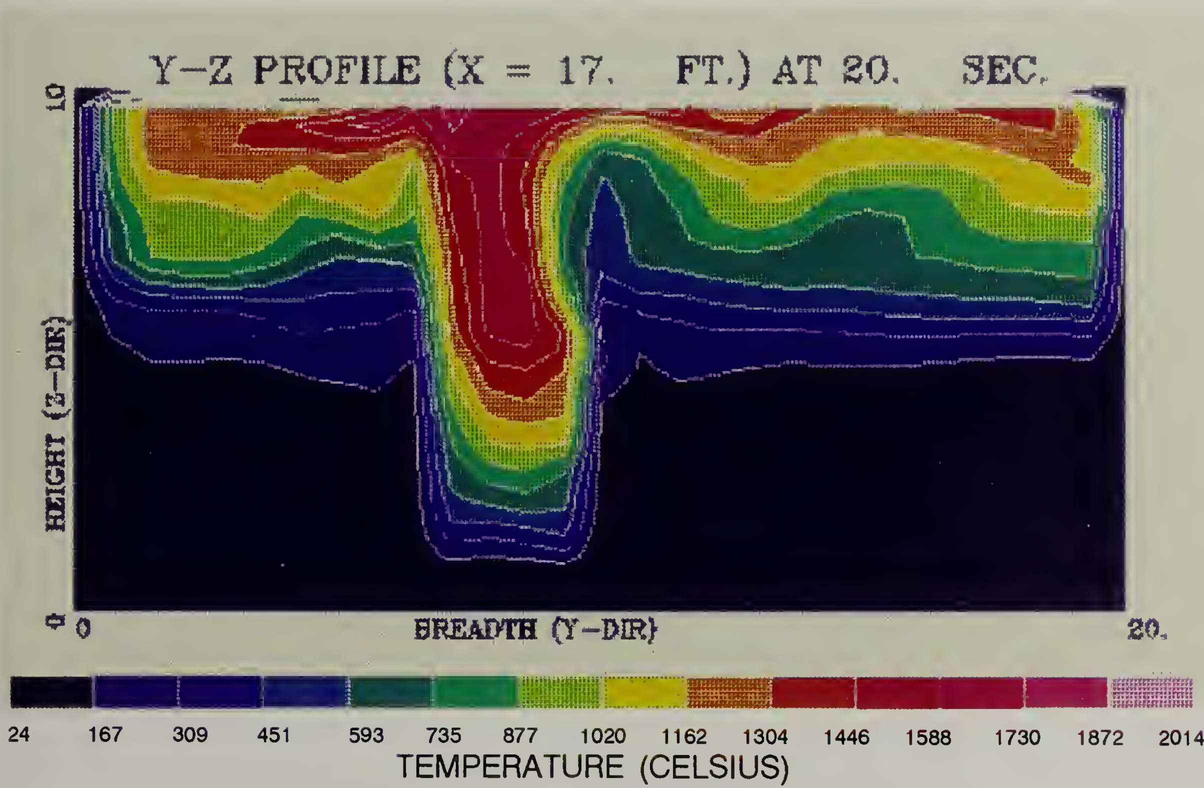


Figure 3.26 Trial 4 Isotherm Profile Elev at 20s

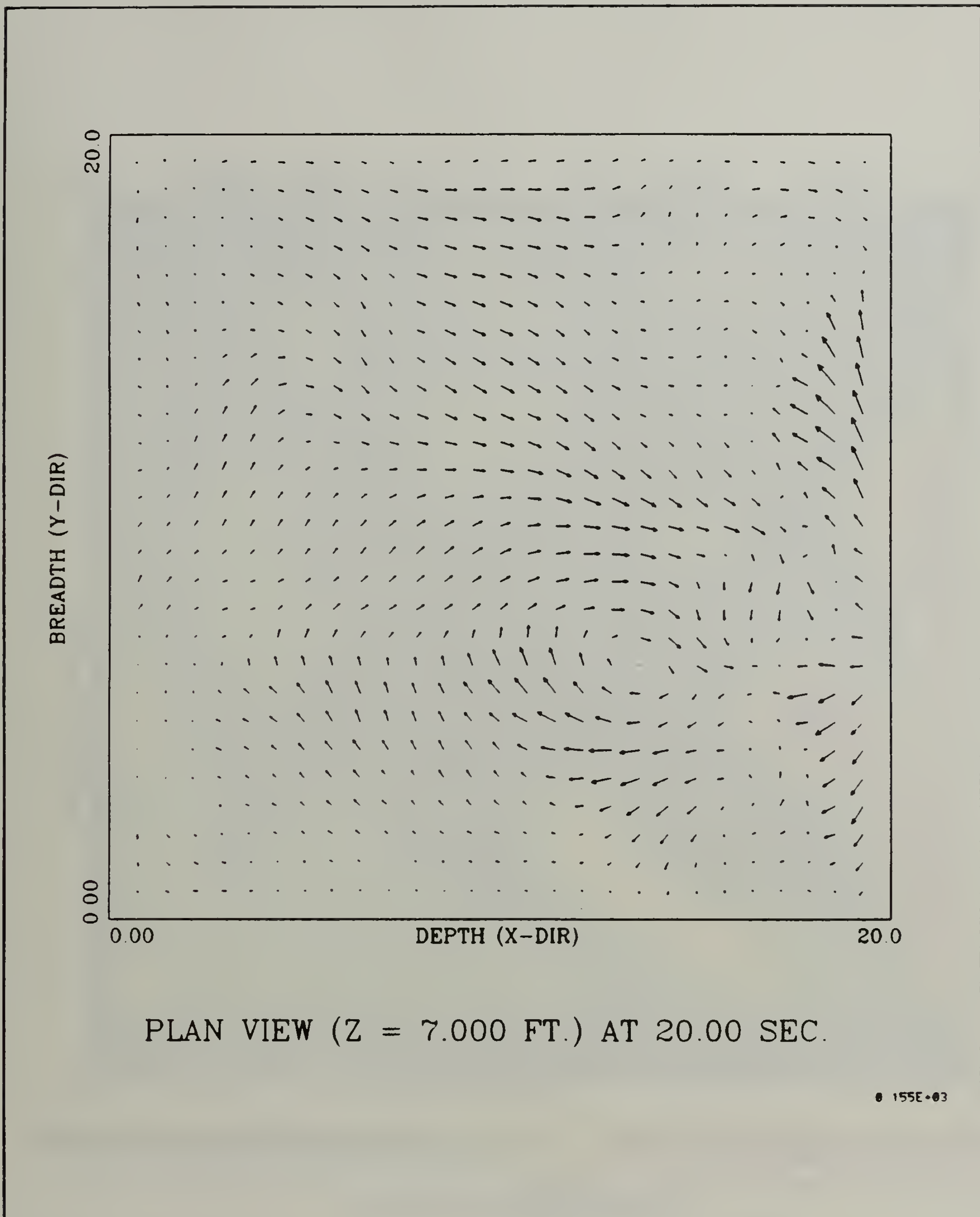


Figure 3.27 Trial 4 Velocity Profile Plan at 20s

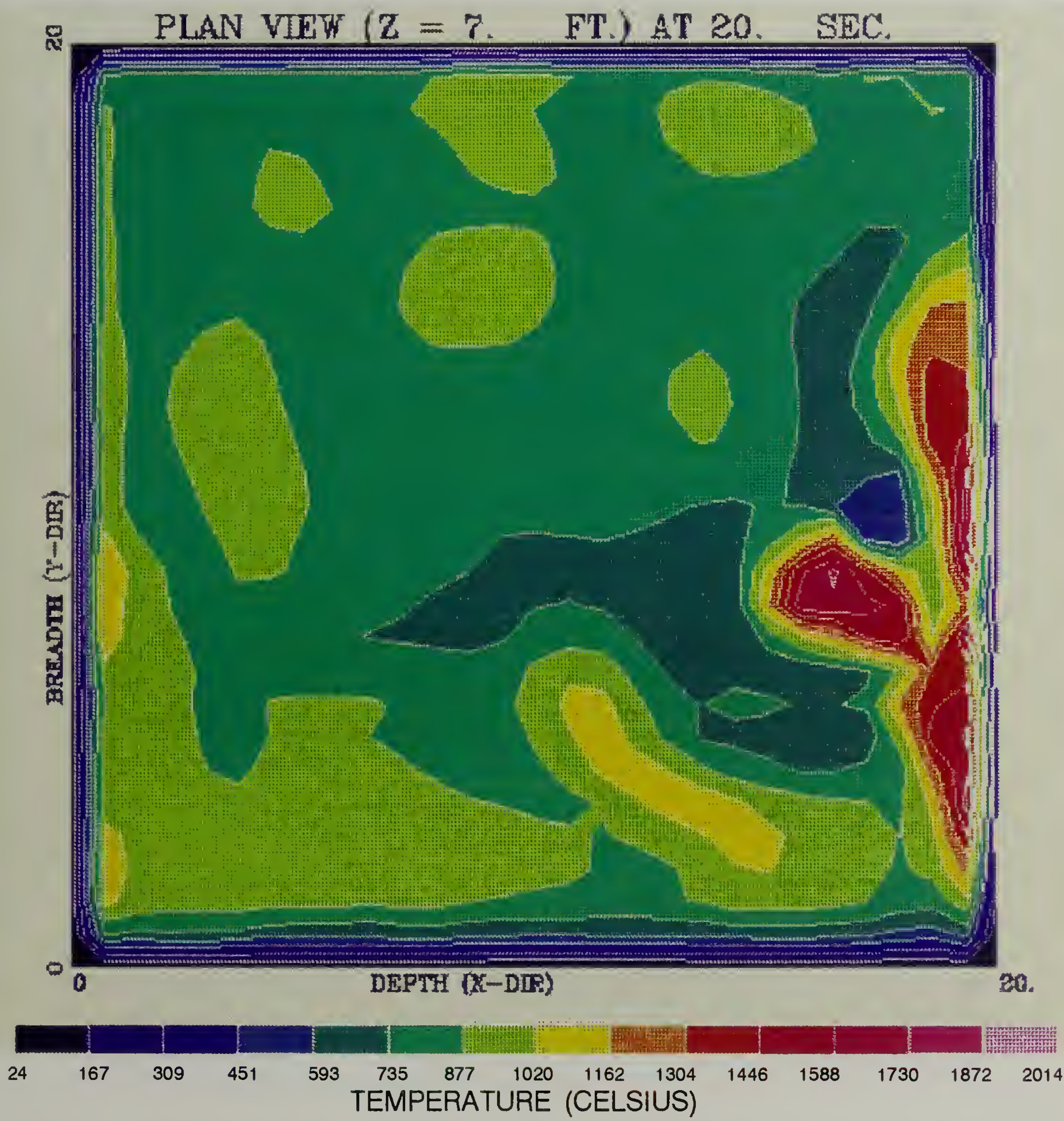


Figure 3.28 Trial 4 Isotherm Profile Plan View at 20s

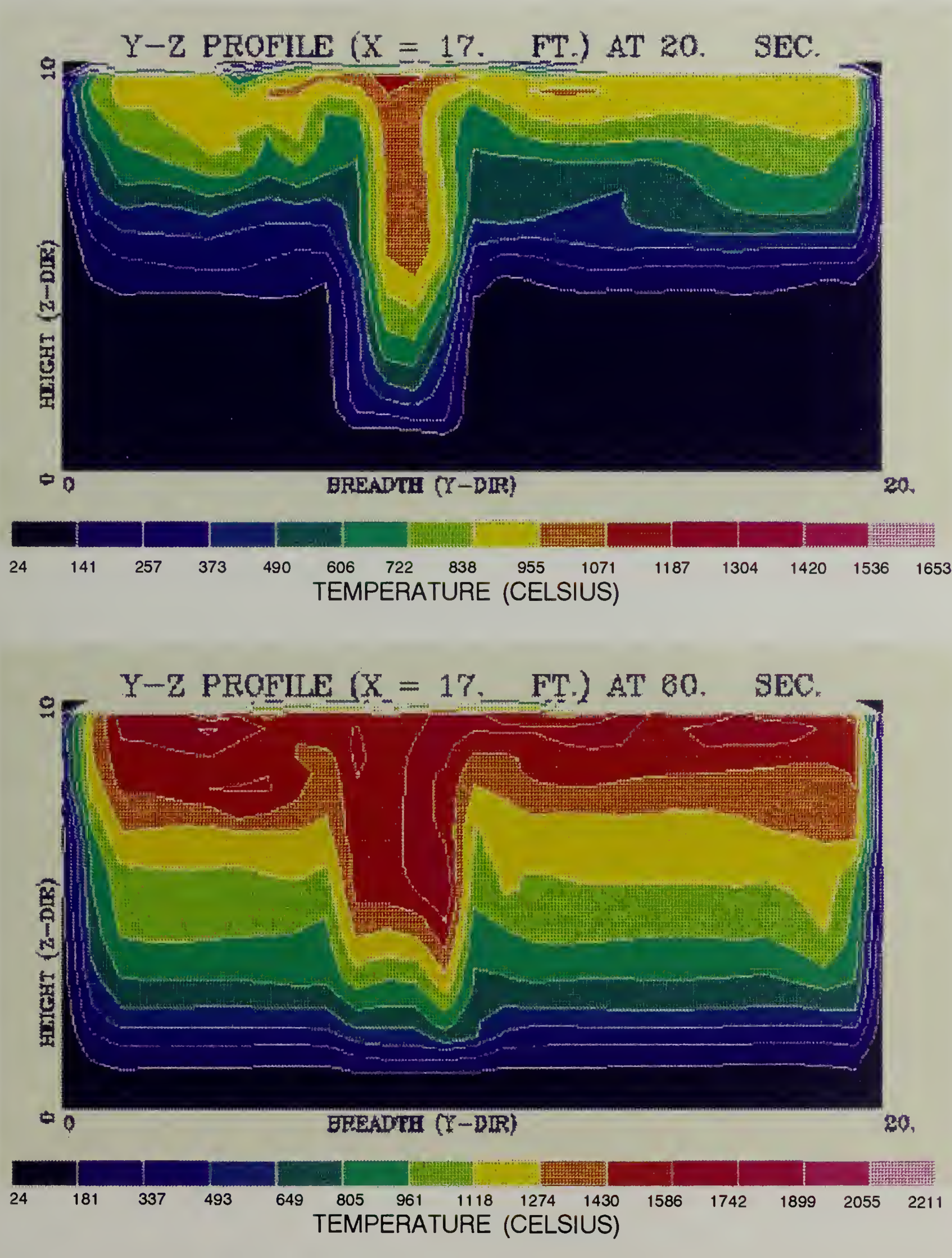


Figure 3.29 Trial 5 Isotherm Profile Elev at 20 and 60s

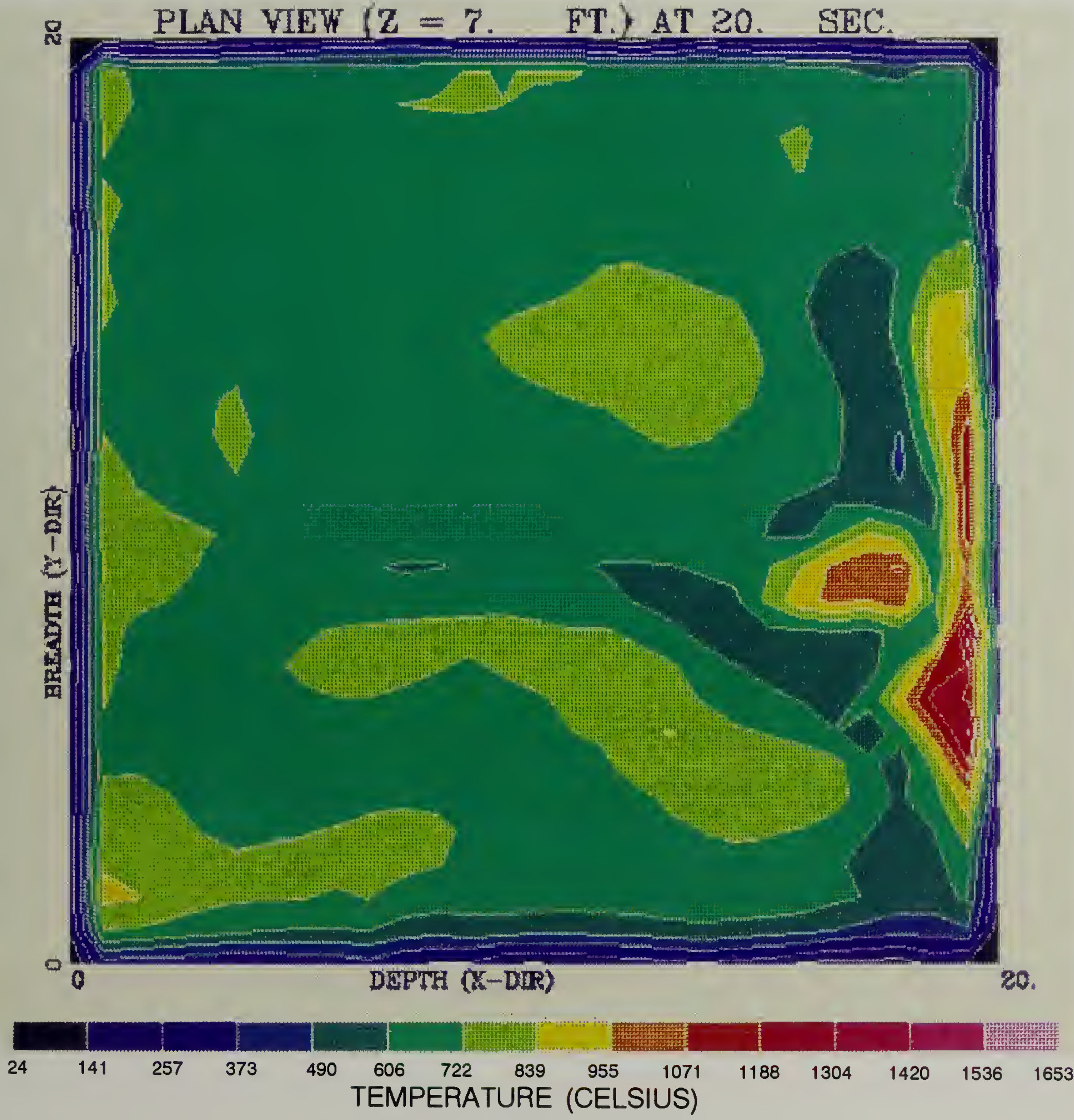


Figure 3.30 Trial 5 Isotherm Profile Plan View at 20s

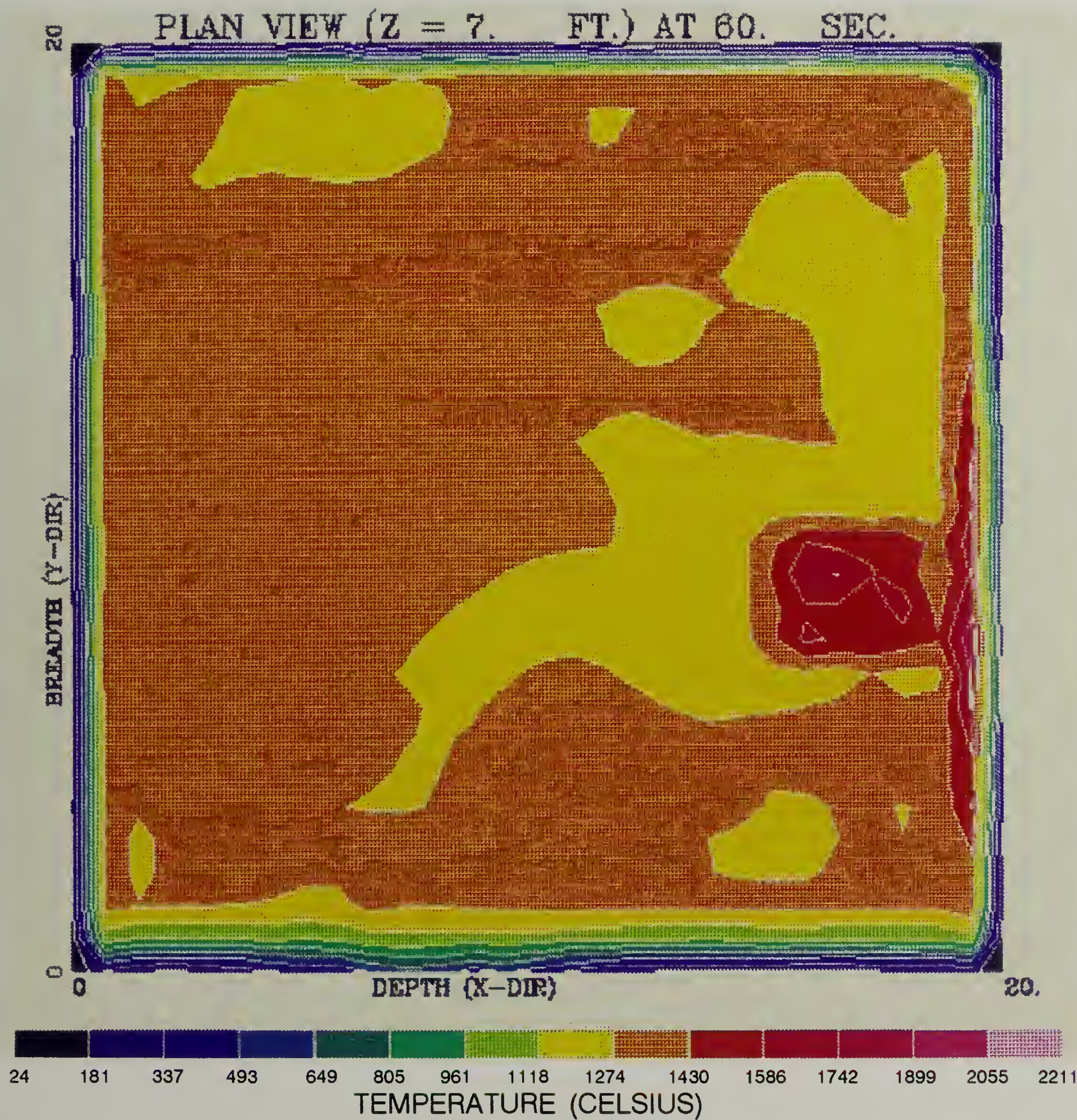


Figure 3.31 Trial 5 Isotherm Profile Plan View at 60s

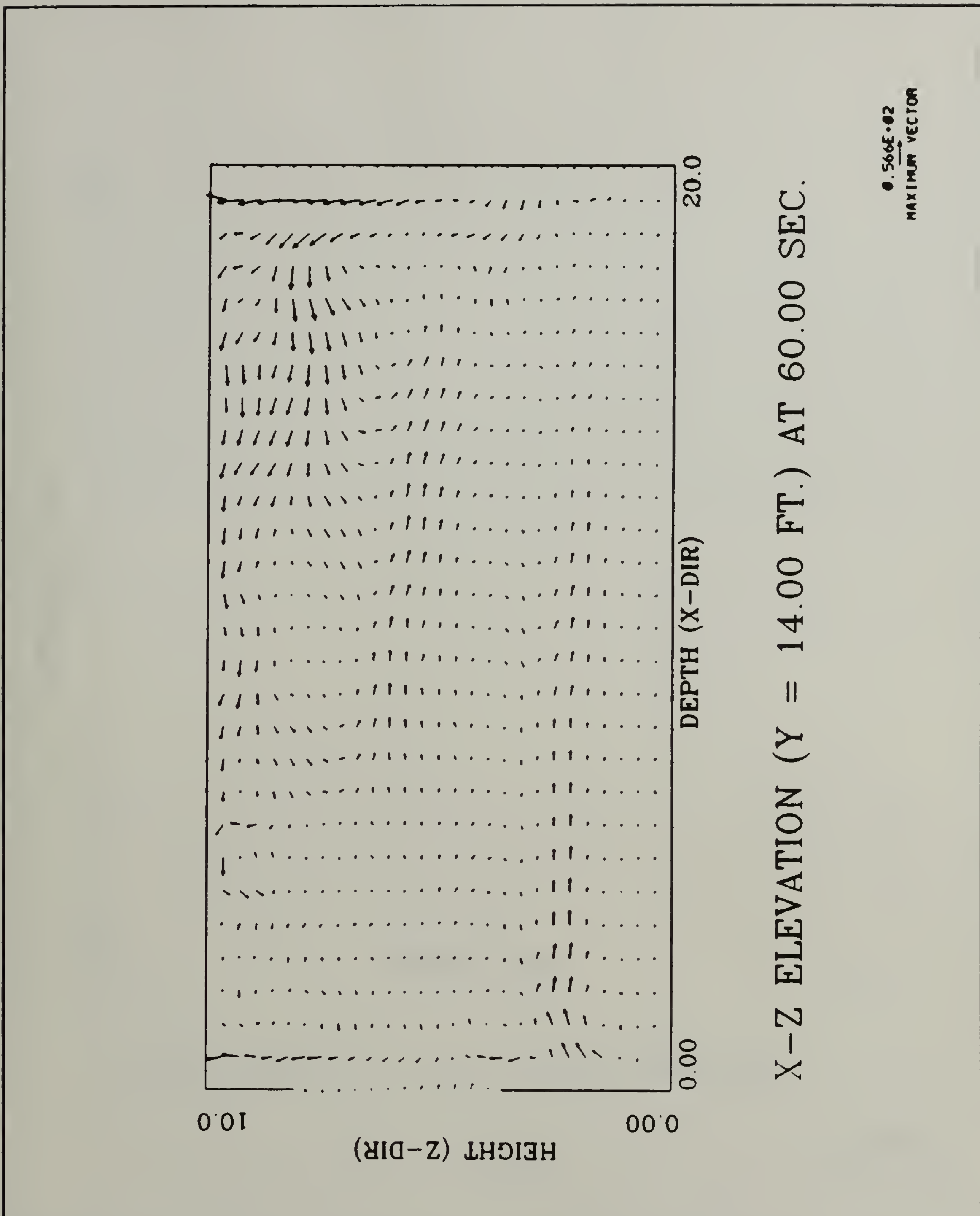


Figure 3.32 Trial 6 Velocity Profile Elev at 20,60s

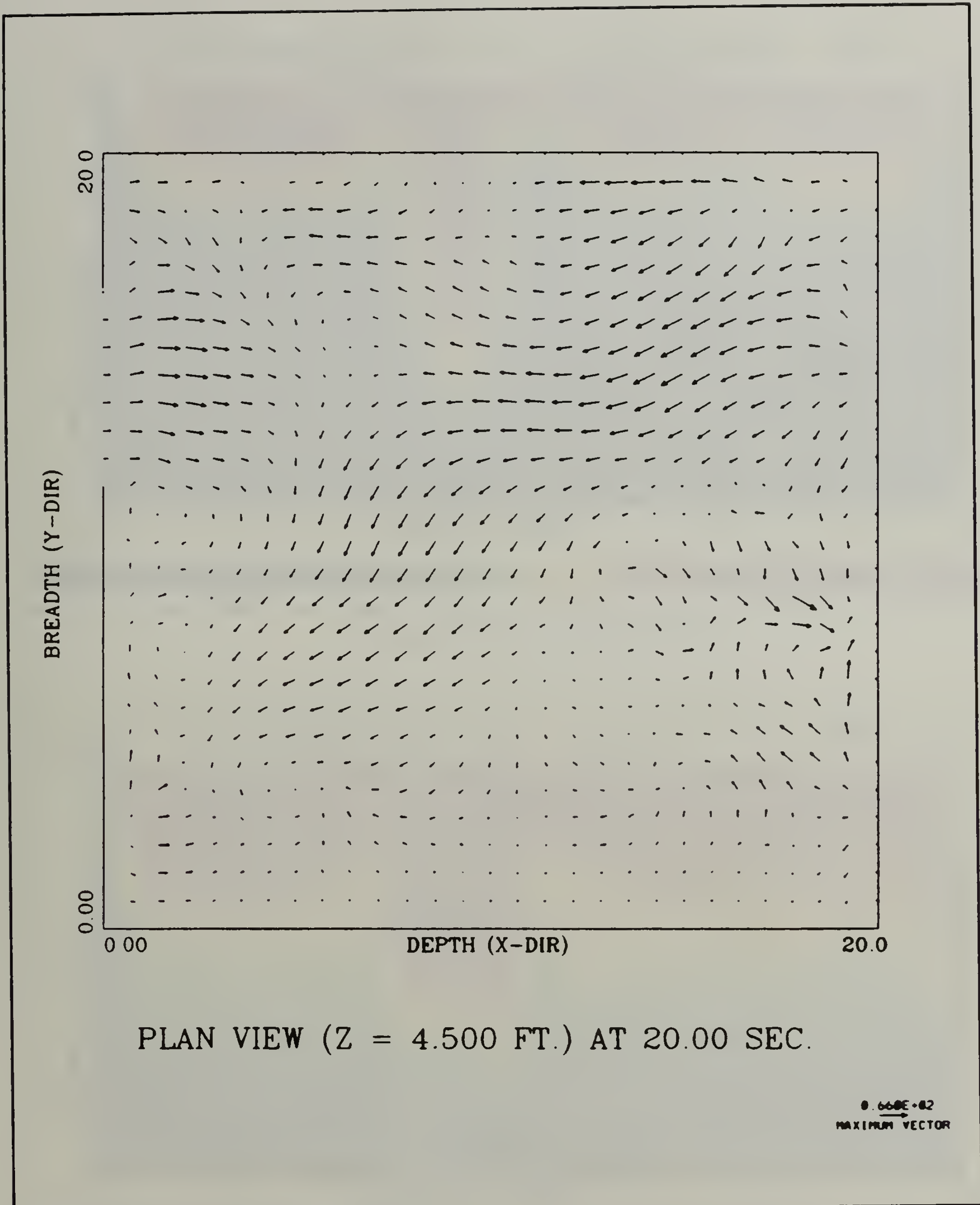
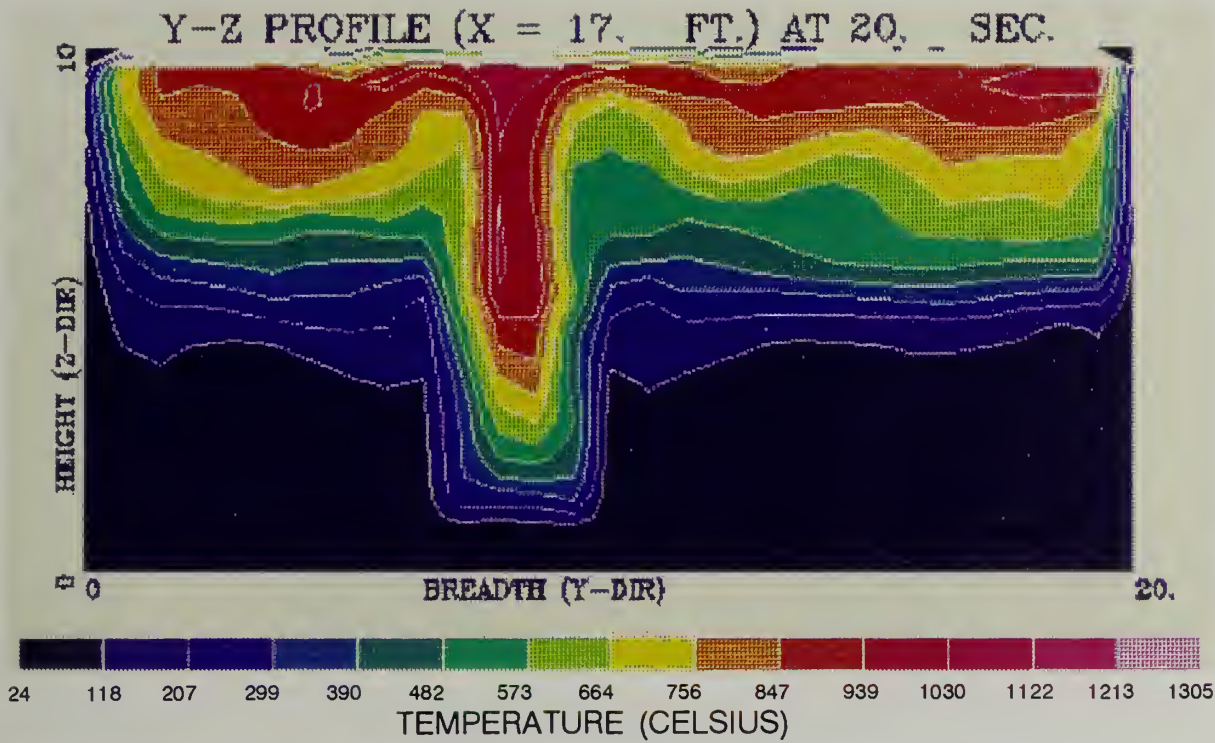


Figure 3.33 Trial 6 Velocity Profile Plan View at 20s



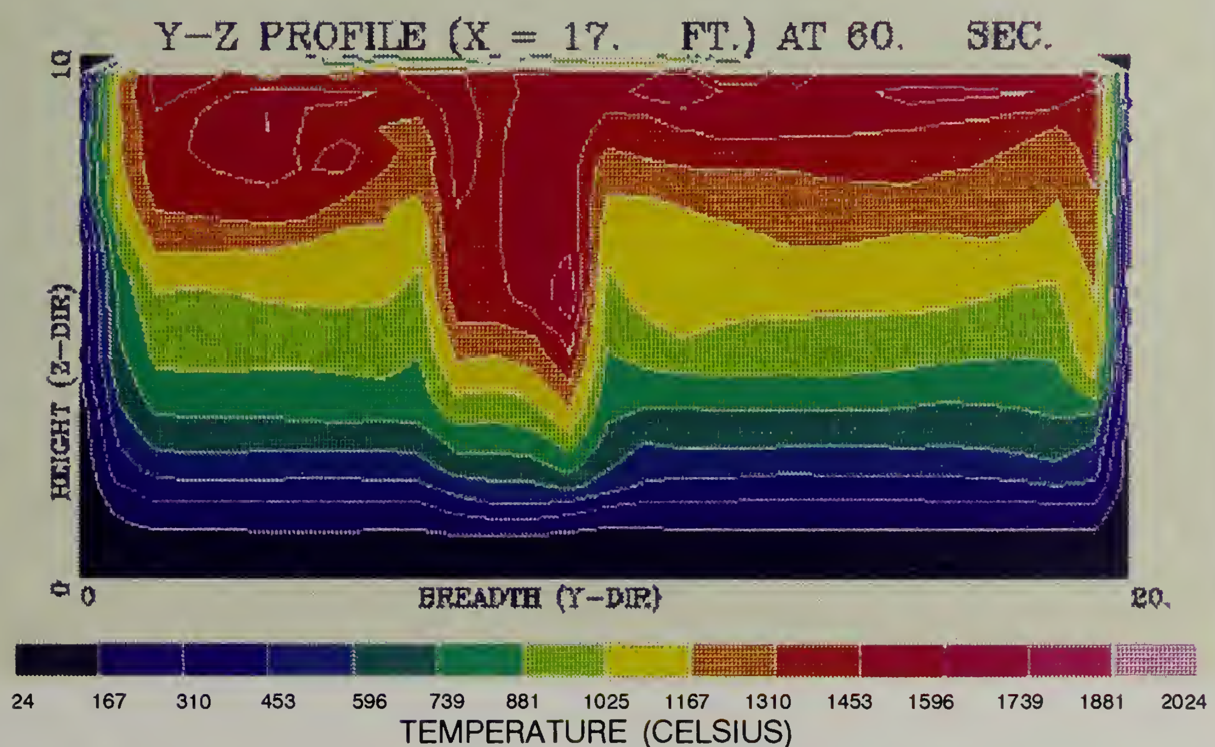


Figure 3.34 Trial 6 Isotherm Profile Elev at 20 and 60s

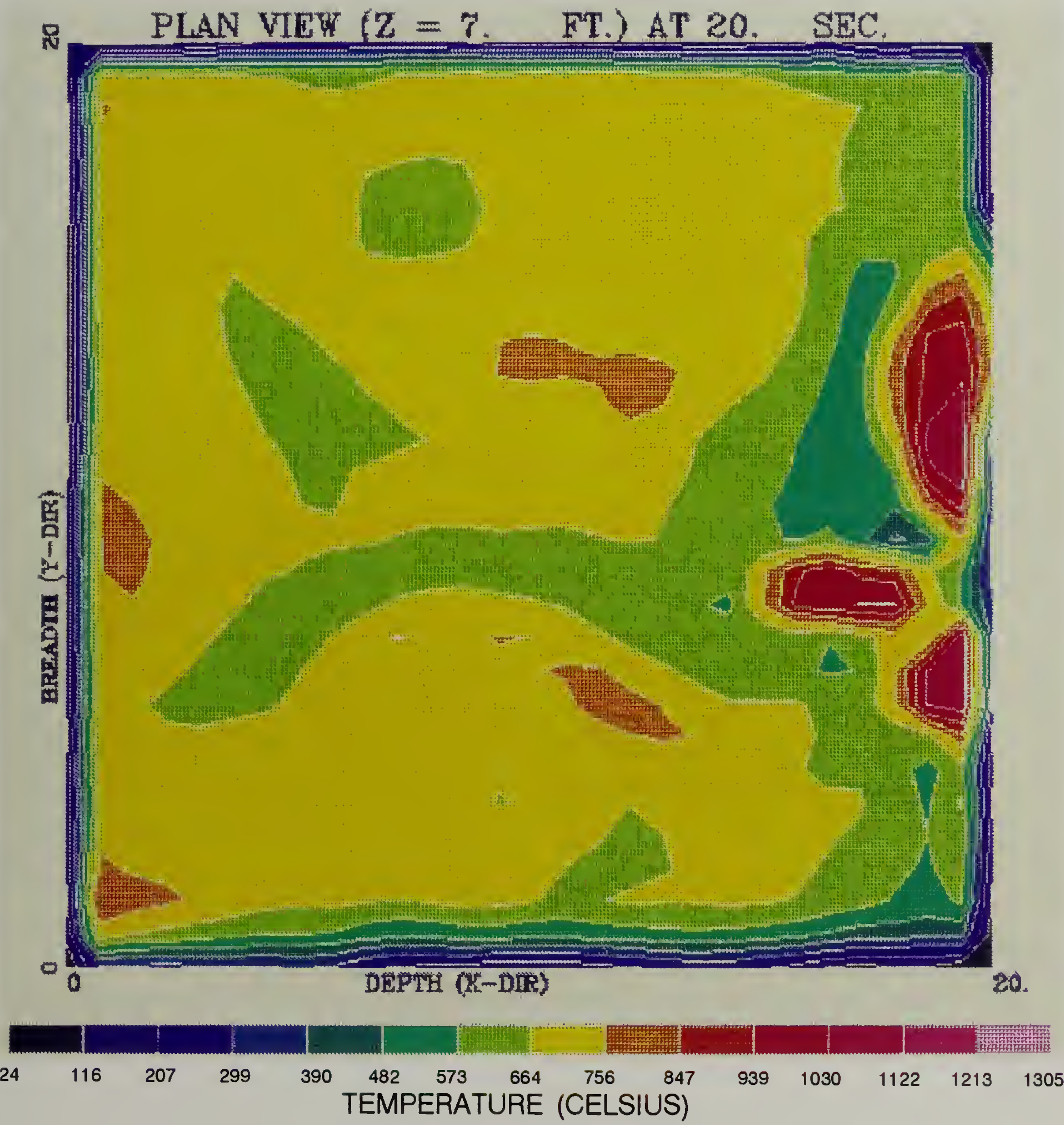


Figure 3.35 Trial 6 Isotherm Profile Plan View at 20s

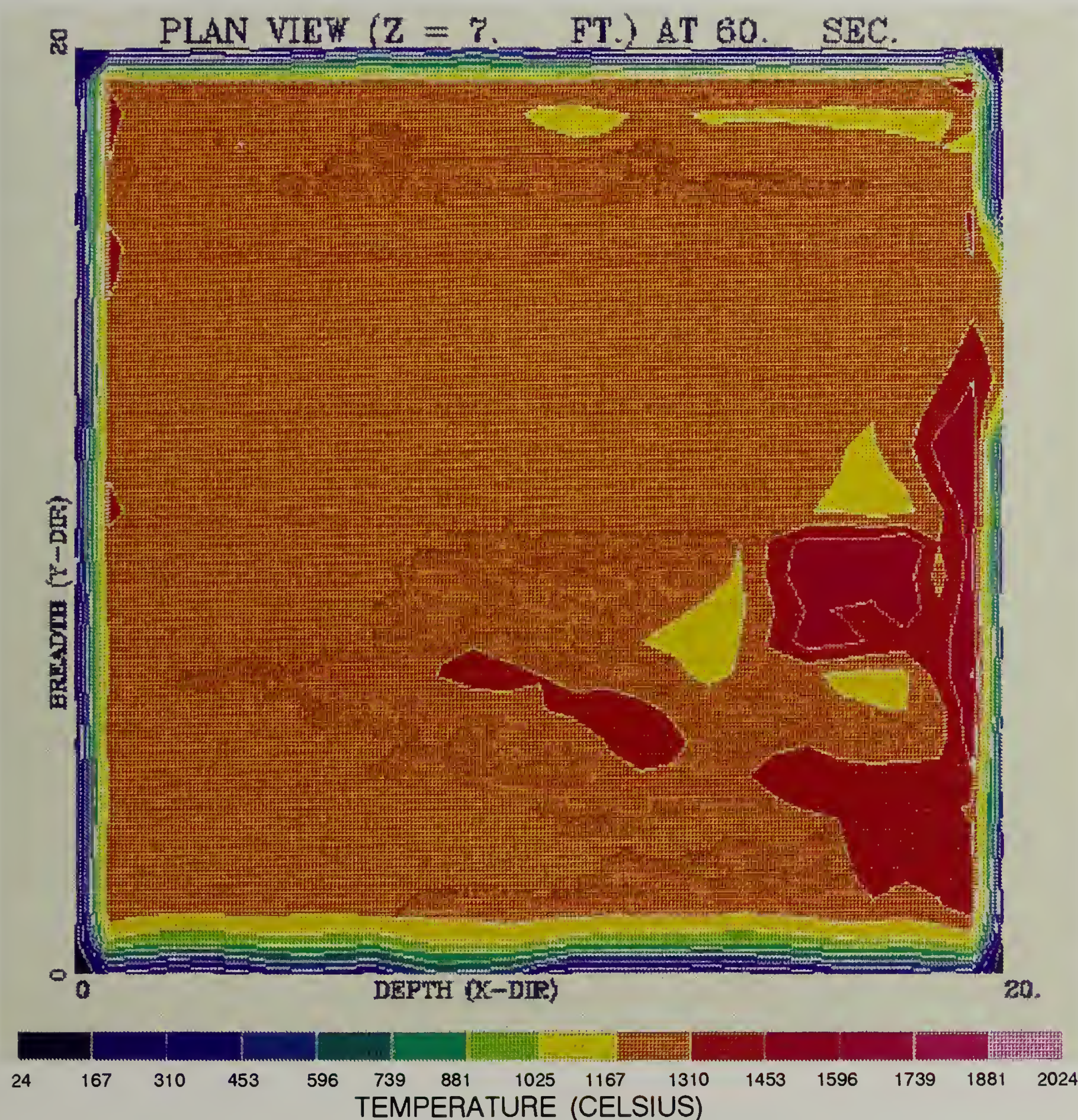


Figure 3.36 Trial 6 Isotherm Profile Plan View at 60s

IV. CONCLUSIONS AND RECOMMENDATIONS

The effort expended in this thesis was fruitful in that it established some guidelines and background for utilizing this program to model a fire from a large magnitude heat source. However, it appears that more questions have been raised than solutions. Deficiencies in the numerical model could be expected to become more evident when subjecting it to a larger scale fire. Stated first are a number of conclusions and then recommendations.

A. CONCLUSIONS

1. Use of a nonuniform grid is necessary. A compromise between computational time and grid size must be made. The runs conducted in this thesis for the closed compartment did not require as many nodes as were used.

2. Time steps should have been increased for these runs.

3. The determination of the overall pressure field affects the solution markedly. This is illustrated by comparison of runs using the global pressure correction routine with those that do not.

4. The problems experienced in this thesis work concerning the conduction model were most likely due to placing the first nodes inside the compartment too far away from the walls. Finer grid refinement in the area adjacent to the walls should correct these problems.

5. Location of the fire affects the solution, as should be expected. The source was placed one foot above the deck and at least two feet away from the walls to ensure that view factors would be calculated to each part of the compartment. Different locations of the fire may require different grid discrectization.

6. Input into the program is in English units. Output is in SI units. The NCAR graphics programs utilize SI units. It would be more convenient to discuss input and output in the same units.

7. The procedure is susceptible to instability given large transients. This was observed in trying to input the maximum heat input over a period of two seconds and again when the source was instantaneously extinguished as in trial one.

8. It should be noted that provision is made for installation of solids in the compartment but no provision is made for their instantaneous combustion.

9. The amount of data that may be output from the program is impressive. The most interesting portions of the fire runs are within the first 20 seconds. The analysis will be valuable because there is an ability to see clearly where hot spots develop and in general, the isotherm plots fit with the velocity plots.

10. View factor and radiation heat transfer routines appear correct. No in depth investigation of these were made. No investigation of the viscosity or stress calculations was conducted.

B. RECOMMENDATIONS

1. The conduction model/interior compartment model interface problem experienced in this thesis need to be resolved. Placing nodes very near the walls should improve the situation.

2. The formulation of the global pressure correction routine for the open compartment should be reviewed.

3. The program includes a smoke generation subroutine. This should eventually be utilized.

4. A maximum cell Peclet number calculation should be incorporated into the scheme. The procedure which appears in appendix C of the program used in Nies' thesis [ref. 20] could serve as guidance.

5. The numerical model results should be compared with empirical data after the resolution of the conduction model problems experienced in this thesis.

APPENDIX A. PROGRAM FLOW CHART

The figures on the following pages provide the program flow chart for the program listed in Appendix B. This flow chart is not entirely complete but the diagrams are accurate to the extent of what they portray.

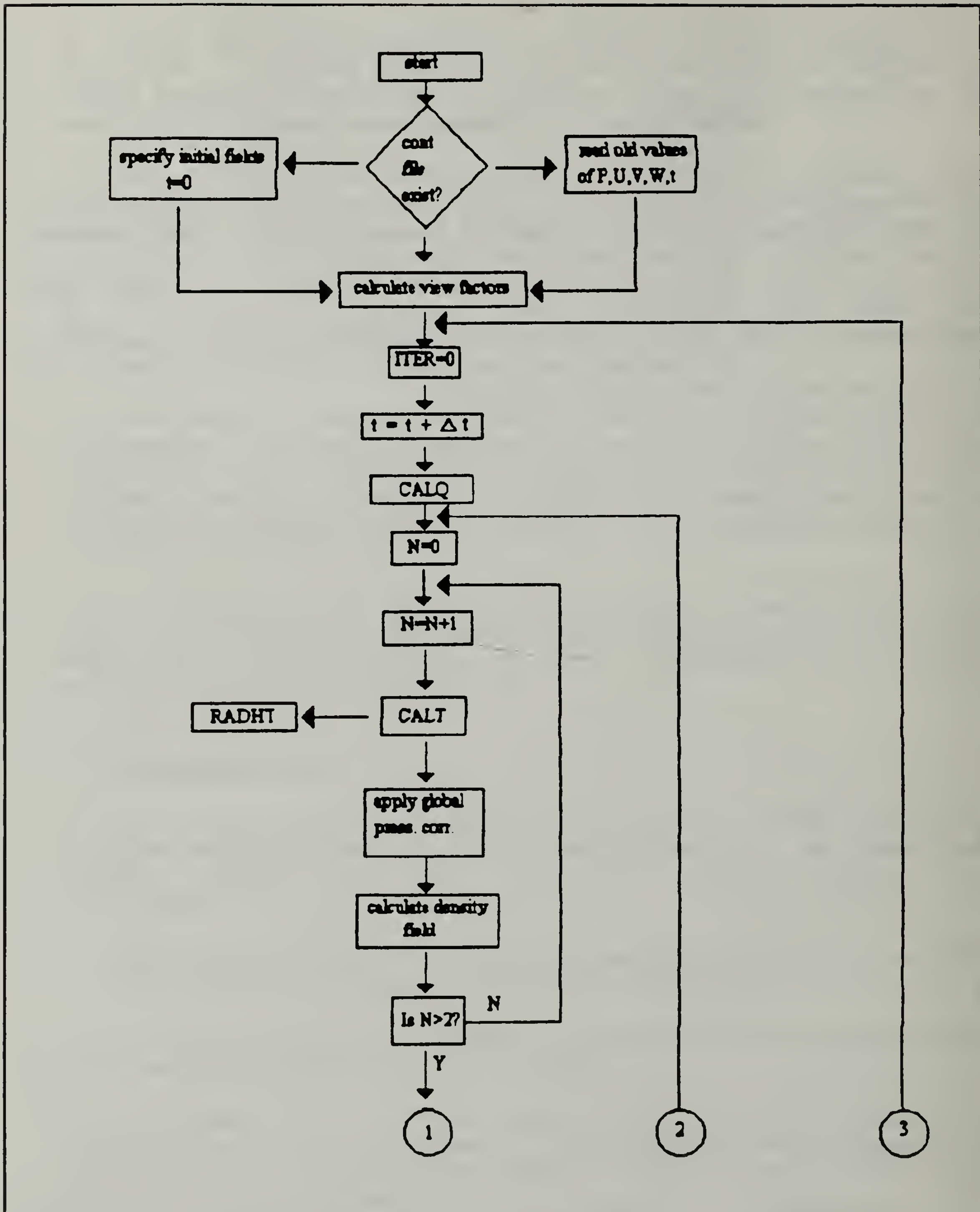


Figure A1 Program Flow Chart

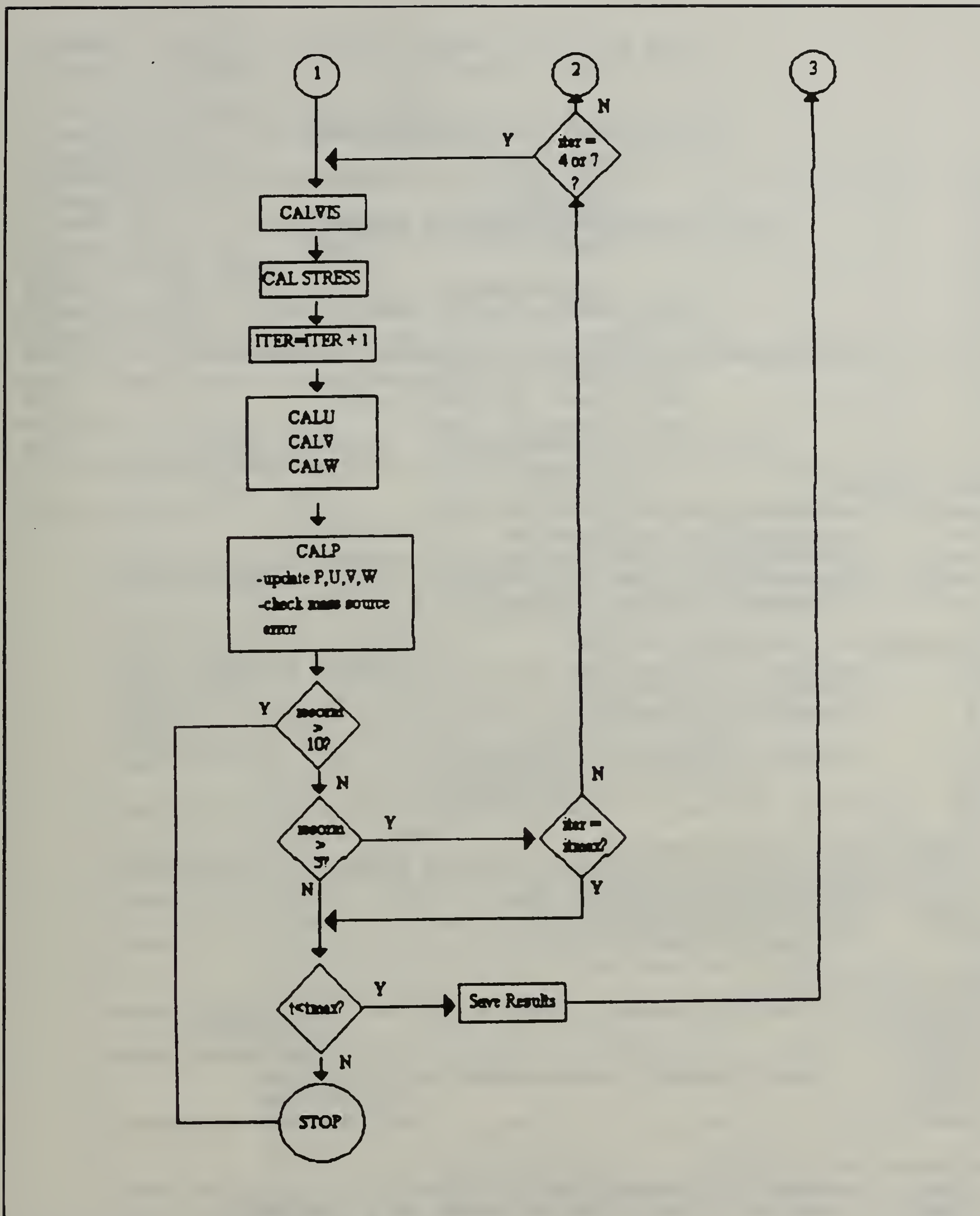


Figure A2 Program Flow Chart

APPENDIX B. PROGRAM FIRE

```
************************************************************************
**                                                                    **
**              THREE-DIMENSIONAL NUMERICAL SIMULATION                **
**                 OF A FIRE SPREAD INSIDE A BUILDING                 **
**                                                                    **
**                          DEVELOPED BY :                            **
**                     H.Q. YANG AND K.T. YANG                        **
**                                                                    **
**         DEPARTMENT OF AEROSPACE & MECHANICAL ENGINEERING           **
**                    UNIVERSITY OF NOTRE DAME                        **
**                   NOTRE DAME, INDIANA, 46556                       **
**                                                                    **
**                            DEC. 1986                               **
************************************************************************
*SET CONSTANTS:
* CP0     : REFERENCE SPECIFIC HEAT OF AIR =
* GC      : GRAVITATIONAL ACCELERATION = 32.17 FT/SEC**2
* RAIR    : UNIVERSAL GAS CONSTANT FOR AIR = 53.34
* RHO0    : REFERENCE AIR DENSITY (LBM/FT**3) = 0.0714 LBM/FT**3
* U0      : REFERENCE VELOCITY (FT/SEC) = 1.0 FT/SEC
*
*COMPARTMENT DIMENSIONS (IN FEET):
* H       : HEIGHT IN Z-DIRECTION (USED AS REFERENCE LENGTH)
* X       : LENGTH IN X-DIRECTION
* Y       : WIDTH IN Y-DIRECTION
*
* NI      : NUMBER OF CELLS IN X-DIRECTION
* NJ      :                    Y-DIRECTION
* NK      :                    Z-DIRECTION
*
* CONSRA  : TA**3/(RA*CP*U0*H*H)
* HCONV   : HEAT TRANSFER COEFFICIENT TO THE AMBIENT (BTU/H*K*FT**2)
* TA      : REFERENCE TEMPERATURE (R)
* TINIT   : INITIAL TEMPERATURE (0)
* UR      : REFERENCE VELOCITY (CM/S)
*
*HEAT SOURCE DATA:
* NHSZ(1,1)  : STARTING CONTROL VOLUME NUMBER IN X-DIRECTION
* NHSZ(2,1)  :                                   Y-DIRECTION
* NHSZ(3,1)  :                                   Z-DIRECTION
* NHSZ(1,2)  : LAST CONTROL VOLUME NUMBER IN X-DIRECTION
* NHSZ(2,2)  :                               Y-DIRECTION
* NHSZ(3,2)  :                               Z-DIRECTION
*
*INTERNAL SOLID PIECES:
* NCHIP    : NUMBER OF INTERNAL SOLID PIECES
* ICHPB()  : STARTING NODE NUMBER FOR SOLID IN X-DIRECTION
* JCHPB()  :                                   Y-DIRECTION
* KCHPB()  :                                   Z-DIRECTION
* NCHPI()  : NUMBER OF NODES OF SOLID IN X-DIRECTION
* NCHPJ()  :                             Y-DIRECTION
* NCHPK()  :                             Z-DIRECTION
*
*TOTAL HEAT:
* QSIN     : INPUT FROM THE FIRE
```

```
* QSWAL    : LOST TO THE WALL (FROM AIR TO THE WALL)
* QSFAN    : CARRIED AWAY BY THE VENTILATION
*
*VIEW FACTORS FROM HEAT SOURCE:
* VFHSW(N,J,K)  : TO ELEMENT J,K ON WEST  WALL
* VFHSE(N,J,K)  :                   EAST  WALL
* VFHSN(N,K,I)  : TO ELEMENT K,I ON NORTH WALL
* VFHSS(N,K,I)  :                   SOUTH WALL
* VFHSF(N,I,J)  : TO ELEMENT I,J ON FRONT WALL
* VFHSB(N,I,J)  :                   BACK  WALL
*VENT OPENING IN WEST WALL:
*NVENT: IF POSITIVE, THERE IS A VENT
*JHWALS,KHWALS: FIRST NODE IN AIR IN Y,Z DIRECTIONS
*JHWALF,KHWALF: FINAL NODE IN AIR IN Y,Z DIRECTIONS
************************************************************************
*DATA FILES USED IN THIS PROGRAM:
*
* FILE # 10 = FIRE DATA      : INITIAL SET-UP DATA
*        11 = CONTINUE DATA  : RESTART/CONTINUATION DATA
*        12 = OUTPUT DATA    : OUTPUT RESULTS
*        13 = PLOT DATA      : DATA FOR PLOTTING
************************************************************************

      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      COMMON/R4/XC(40),YC(40),ZC(40),XS(40),YS(40),ZS(40),DXXC(40),
     &          DYYC(40),DZZC(40),DXXS(40),DYYS(40),DZZS(40)
      COMMON/BL1/DX,DY,DZ,DTIME,TCOOL,PI,Q,QR
      COMMON/BL2/X,Y,H,TFLR,TWAL
      COMMON/BL3/F,FR,HSTART
      COMMON/BL7/NI,NJ,NK,KRUN,NBLOR,NWRP
      COMMON/BL12/NWRITE,NTAPE,NTMAX0,NTREAL,TIME,SORSUM,ITER
      COMMON/BL14/HCOEF,CNT,ABTURB,BTURB,VISL,VISMAX
      COMMON/BL16/U0,UGRT,BUOY,CP0,PRT,COND0,VIS0,RHO0,
     &          TA,DTEMP,TWRITE,TTAPE,TMAX,GC,RAIR,NT
      COMMON/BL23/RMS(20),NMS,IMSB(20),NMSI(20),JMSB(20),NMSJ(20),
     &          KMSB(20),NMSK(20)
      COMMON/BL20/SIG11(30,30,25),SIG12(30,30,25),SIG22(30,30,25),
     &          SIG13(30,30,25),SIG23(30,30,25),SIG33(30,30,25)
      COMMON/BL22/CPS(20),CONS(20),WFAN(20),NCHIP,ICHPB(20),NCHPI(20),
     &          JCHPB(20),NCHPJ(20),KCHPB(20),NCHPK(20)
      COMMON/BL31/TOD(30,30,25),ROD(30,30,25),POD(30,30,25),
     &          COD(30,30,25),UOD(30,30,25),VOD(30,30,25),
     &          WOD(30,30,25)
      COMMON/BL32/T(30,30,25),R(30,30,25),P(30,30,25),C(30,30,25),
     &          U(30,30,25),V(30,30,25),W(30,30,25)
      COMMON/BL33/TPD(30,30,25),RPD(30,30,25),PPD(30,30,25),
     &          CPD(30,30,25),UPD(30,30,25),VPD(30,30,25),
     &          WPD(30,30,25)
      COMMON/BL34/HEIGHT(30,30,25),REQ(30,30,25),SMP(30,30,25),
     &          SMPP(30,30,25),PP(30,30,25),DU(30,30,25),
     &          DV(30,30,25),DW(30,30,25)
      COMMON/BL36/AP(30,30,25),AE(30,30,25),AW(30,30,25),AN(30,30,25),
     &          AS(30,30,25),AF(30,30,25),AB(30,30,25),SP(30,30,25),
     &          SU(30,30,25),RI(30,30,25)
      COMMON/BL37/VIS(30,30,25),COND(30,30,25),RESORM(40),
     &          CPM(30,30,25),NHSZ(3,2),NOD(30,30,25)
      COMMON/BL38/TCOUP(30),CX(30),CY(30),CZ(30),NTH(30,3),NTHCO
      COMMON/BL39/ALEW,CONSRA,QSIN,QSWER,QSWAL,QSAIR,QSFAN,QSHOL
      COMMON/BL40/VFHSW(8,30,30),VFHSE(8,30,30),VFHSS(8,30,30),
     &          VFHSN(8,30,30),VFHSB(8,30,30),VFHSF(8,30,30)
      COMMON/BL41/VFHSBW(8,8,34,34),VFHSBE(8,8,34,34),VFHSBS(8,8,34,34),
```

```
     &              VFHSBN(8,8,34,34),VFHSBB(8,8,34,34),VFHSBF(8,8,34,34)
      COMMON/BL43/QSCONF,QSCONB,QSCONE,QSCONW,QSCONN,QSCONS,QSCONH,
     &              QSRADF, QSRADB,QSRADE,QSRADW,QSRADN,QSRADS,QSRHOL,
     &              WAIR,WWAL,WINS,WERR,WWFAN,WWHOL
      COMMON/BL50/JHWALS,JHWALF,KHWALS,KHWALF,NVENT
      COMMON/BL51/SSGPW(20,20,20)
      COMMON/BL52/CONVMT
      DATA SORMAX,XTIME,ITMAX/3.00,0.0,4/
      SOURCE=0.0
      ITERT=0.0
C ***** INITIAL PROGRAM START *****
      CALL CPUTIME(BEGIN,IPR)

C *** INPUT DATA
      CALL INPUT(NSTOP)
      IF(NSTOP.GT.0) GOTO 9999
C *** GENERATE GRID SYSTEM
      CALL GRID

C *** INITIALIZE THE ALL DATA FIELDS
      CALL INIT

C *** OPEN OUTPUT FILES
C *** FILES ARE WRITTEN TO DIFFERENT DISKS
      OPEN(12,FILE='/OUTPUT DATA D1',STATUS='UNKNOWN')
      OPEN(13,FILE='/PLOT DATA B4',STATUS='UNKNOWN',
     &        FORM='UNFORMATTED')

C *** CALCULATE THE VIEW FACTORS FROM THE FIRE TO THE WALLS
      CALL VIEW

***** START CALCULATIONS *****

      NT=0
      NTIM=0
  300 NT=NT+1

C *** ON RESTART NTMAX0 IS SET EQUAL TO OLD VALUE FOR NTREAL
      IF(TIME.GE.TMAX) GO TO 277
      NTREAL=NT+NTMAX0
      TIME=TIME+DTIME
      XTIME=TIME*H/U0
      PRINT 3,'CURRENT FIRE TIME IS:',XTIME,'SECONDS'
    3 FORMAT (1X,A,1X,F10.6,1X,A)
C *** CALCULATE THE HEAT SOURCE IN BTU/SEC
      CALL CALQ

C *** START CALCULATIONS
      ITER=0
      JTERM=0
      JJTERM=0

C *** PREDICT VARIABLE FIELDS FOR USE BY CALVIS AND SU(I,J,K)
      DO 48 K=1,NK+4
      DO 48 J=1,NJ+4
      DO 48 I=1,NI+4
         TPD(I,J,K)=T(I,J,K)
         CPD(I,J,K)=C(I,J,K)
         RPD(I,J,K)=R(I,J,K)
         UPD(I,J,K)=U(I,J,K)
         VPD(I,J,K)=V(I,J,K)
```

```
         WPD(I,J,K)=W(I,J,K)
   48 CONTINUE
   47 JTERM=JTERM+1
  301 NTITER=0
  312 NTITER=NTITER+1
C *** IF FIRE HAS STARTED, CALCULATE THE TEMPERATURE
      IF (XTIME.GE.HSTART) CALL CALT

*****THIS STEP CAN BE SKIPPED WHEN COMPARTMENT IS OPEN TO OUTSIDE*****
C *** CORRECT GLOBAL PRESSURE FOR TOTAL MASS CONSERVATION
      CALL GLOBE

C *** CALCULATE DENSITY
      DO 100 J=1,NJ+4
      DO 100 I=1,NI+4
      DO 100 K=1,NK+4
         IF (NOD(I,J,K).EQ.1) GOTO 100
         AAAA=BUOY*UGRT*HEIGHT(I,J,K)
         R(I,J,K)=(UGRT*P(I,J,K)+(1./EXP(AAAA)))/T(I,J,K)
  100 CONTINUE

C *** ITERATE INSIDE TEMPERATURE LOOP TO ASSURE GLOBAL CONSERVATION
C *** OF MASS AND ENERGY
      IF (NTITER.LT.2) GOTO 312

C *** PRINT OUT THE ENERGY DISTRIBUTION
       IF (MOD(NTREAL,NWRP).EQ.0) CALL OUT(4)

C *** CALCULATE THE SMOKE CONCENTRATION
C     CALL CALC

C *** CALCULATE TURBULENT VISCOSITY AND CONDUCTIVITY
      CALL CALVIS

C *** CORRECT CONDUCTIVITY OF THE SOLID
      IF (NCHIP.NE.0) CALL SOLCON

C *** START PRESSURE CORRECTION ITERATIVE LOOP
C *** IT IS THE MAJOR PART OF THE ERROR CONTROL ROUTINE
      ITER=ITER+1

C *** CALCULATE THE STRESS AND VELOCITY COMPONENTS U,V,AND W
      CALL STRESS
      CALL CALU
      CALL CALV
      CALL CALW


C *** CALCULATE PRESSURE
      CALL CALP

      PRINT *,'RESORM=',RESORM(ITER)


C *** IF SOURCE TERM IS LARGER THAN 10.0, STOP PROGRAM
      IF (RESORM(ITER).GT.10.0) GOTO 2020

      IF(RESORM(ITER).LE.SORMAX) GO TO 49
      IF(ITER.EQ.1) GO TO 302
      IF(RESORM(ITER) .LE. RESORM(ITER-1)) GO TO 302
      GO TO 304
```

```
302 IF(JTERM .LT. 2) THEN
       SOURCE=RESORM(ITER)
    ELSEIF(RESORM(ITER).LE.SOURCE) THEN
       SOURCE=RESORM(ITER)
    ELSE
       GOTO 304
    ENDIF
    DO 23 K=1,NK+4
    DO 23 J=1,NJ+4
    DO 23 I=1,NI+4
       TPD(I,J,K)=T(I,J,K)
       CPD(I,J,K)=C(I,J,K)
       RPD(I,J,K)=R(I,J,K)
       UPD(I,J,K)=U(I,J,K)
       VPD(I,J,K)=V(I,J,K)
       WPD(I,J,K)=W(I,J,K)
       PPD(I,J,K)=P(I,J,K)
 23 CONTINUE

    JJTERM=0
    IF(ITER.EQ.ITMAX) GO TO 49
    IF(JTERM.EQ.2) GO TO 35
    IF(ITER.EQ.4) GO TO 47
 35 IF(JTERM.EQ.3) GO TO 58
    IF(ITER.EQ.7) GO TO 47
 58 JJTERM=0
    GO TO 301
304 JJTERM=JJTERM+1
    IF(JTERM.EQ.1) GOTO 41
    IF(JTERM.EQ.2.AND.JJTERM.EQ.1.AND.ITER.NE.5) GO TO 41
    GO TO 82

 41 DO 40 K=1,NK+4
    DO 40 J=1,NJ+4
    DO 40 I=1,NI+4
       R(I,J,K)=RPD(I,J,K)
       U(I,J,K)=UPD(I,J,K)
       V(I,J,K)=VPD(I,J,K)
       W(I,J,K)=WPD(I,J,K)
       P(I,J,K)=PPD(I,J,K)
 40 CONTINUE
    IF(ITER.EQ.ITMAX) GO TO 49
    GO TO 47

 82 DO 43 K=1,NK+4
    DO 43 J=1,NJ+4
    DO 43 I=1,NI+4
       T(I,J,K)=TPD(I,J,K)
       C(I,J,K)=CPD(I,J,K)
       R(I,J,K)=RPD(I,J,K)
       U(I,J,K)=UPD(I,J,K)
       V(I,J,K)=VPD(I,J,K)
       W(I,J,K)=WPD(I,J,K)
       P(I,J,K)=PPD(I,J,K)
 43 CONTINUE

    IF(ITER.EQ.ITMAX) GO TO 49
    IF((JTERM.EQ.3.AND.ITER.NE.8).OR.JJTERM.EQ.2) GO TO 49
    GO TO 301

 49 ITERT=ITERT+ITER
```

```
C      IF (MOD(NTREAL,NWRP).EQ.0) CALL OUT(1)

C *** FIND TEMPERATURES AT THERMOCOUPLES AND PRINT OUT AT PROPER TIME
      CALL TCP
      IF (MOD(NTREAL,NWRITE).EQ.0) CALL OUT(2)

C *** OUTPUT FILED VALUES
      IF (MOD(NTREAL,NWRITE).EQ.0) CALL OUT(3)
      IF(TIME.GE.TMAX) GO TO 277

C *** SHIFT CURRENT TIME VALUES TO PREVIOUS TIME VALUES AND
C *** LOOP BACK FOR NEXT ITERATION
      DO 305 K=1,NK+4
      DO 305 J=1,NJ+4
      DO 305 I=1,NI+4
         TOD(I,J,K)=T(I,J,K)
         COD(I,J,K)=C(I,J,K)
         ROD(I,J,K)=R(I,J,K)
         UOD(I,J,K)=U(I,J,K)
         VOD(I,J,K)=V(I,J,K)
         WOD(I,J,K)=W(I,J,K)
         POD(I,J,K)=P(I,J,K)
  305 CONTINUE

C *** OUTPUT TO DATA FILE FOR PLOTTING
      IF(MOD(NTREAL,NTAPE).EQ.0) THEN
         WRITE(13) TIME,T,U,V,W
      ENDIF

C *** OUTPUT TO CONTINUATION FILE FOR RESTART
      IF(MOD(NTREAL,200).EQ.0) THEN
         OPEN(11,FILE='/CONTINUE DATA B4',STATUS='UNKNOWN',
     &           FORM='UNFORMATTED')
         WRITE(11) TIME,NTREAL,CONVMT,FR,T,R,U,V,W,P,C
         REWIND 11
         CLOSE(11)
      ENDIF

      GO TO 300

C *** OUTPUT TO CONTINUATION FILE
  277    OPEN(11,FILE='/CONTINUE DATA B4',STATUS='UNKNOWN',
     &           FORM='UNFORMATTED')
         WRITE(11) TIME,NTREAL,CONVMT,FR,T,R,U,V,W,P,C
         REWIND 11
         CLOSE(11)
      GO TO 9999

 2020 WRITE(12,*)'RESIDUAL MASS IS LARGER THAN 10.0',
     &           ' PROGRAM STOPS AT TIME = ',XTIME,' SEC'
      WRITE(12,*)'RESORM=',RESORM(ITER)
 9999 CALL CPUTIME(END,IPR)
      WRITE (12,*) 'CPU RUN TIME = ',(END-BEGIN)*1.E6,' SECONDS'
      STOP
      END

************************************************************************
************************************************************************
      BLOCK DATA
************************************************************************
* U0     : REFERENCE VELOCITY = 1.0 FT/SEC
```

```
* PRT    : TURBULENT PRANDTL NUMBER = 1.0
* RHO0   : REFERENCE DENSITY OF AIR = 0.0714 LBM/FT**3
* CP0    : REFERENCE SPECIFIC HEAT OF AIR = 0.24 BTU/(LBM*F)
* VIS0   : REFERENCE VISCOSITY = 1.56E-4
* CNT    :
* ABTURB : TURBULENCE CONSTANT
* BTURB  : TURBULENCE CONSTANT
* GC     : GRAVITATIONAL ACCELERATION = 32.17 FT/SEC**2
* RAIR   : GAS CONSTANT FOR AIR = 53.34
* ALEW   : LEWIS NUMBER = 1.0
************************************************************************

      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      COMMON/BL12/NWRITE,NTAPE,NTMAX0,NTREAL,TIME,SORSUM,ITER
      COMMON/BL14/HCOEF,CNT,ABTURB,BTURB,VISL,VISMAX
      COMMON/BL16/U0,UGRT,BUOY,CP0,PRT,COND0,VIS0,RHO0,
     &            TA,DTEMP,TWRITE,TTAPE,TMAX,GC,RAIR,NT
      COMMON/BL39/ALEW,CONSRA,QSIN,QSWER,QSWAL,QSAIR,QSFAN,QSHOL

C *** SPECIFY THE INITIAL DATA
      DATA  U0, PRT,  RHO0 ,  CP0,    VIS0  , NTMAX0/
     &     1.0, 1.0, 0.0714, 0.24,  1.56D-4,    0/
      DATA CNT,ABTURB,BTURB/0.2,2.0,1.0/
      DATA GC,RAIR,ALEW/32.17,53.34,1.0/

      END

************************************************************************
************************************************************************
      SUBROUTINE CALC
************************************************************************
*THIS SUBROUTINE CALCULATES THE SMOKE CONCENTRATIONS

      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      COMMON/R4/XC(40),YC(40),ZC(40),XS(40),YS(40),ZS(40),DXXC(40),
     &          DYYC(40),DZZC(40),DXXS(40),DYYS(40),DZZS(40)
      COMMON/BL1/DX,DY,DZ,DTIME,TCOOL,PI,Q,QR
      COMMON/BL2/X,Y,H,TFLR,TWAL
      COMMON/BL7/NI,NJ,NK,KRUN,NBLOR,NWRP
      COMMON/BL16/U0,UGRT,BUOY,CP0,PRT,COND0,VIS0,RHO0,
     &            TA,DTEMP,TWRITE,TTAPE,TMAX,GC,RAIR,NT
      COMMON/BL31/TOD(30,30,25),ROD(30,30,25),POD(30,30,25),
     &            COD(30,30,25),UOD(30,30,25),VOD(30,30,25),
     &            WOD(30,30,25)
      COMMON/BL32/T(30,30,25),R(30,30,25),P(30,30,25),C(30,30,25),
     &            U(30,30,25),V(30,30,25),W(30,30,25)
      COMMON/BL33/TPD(30,30,25),RPD(30,30,25),PPD(30,30,25),
     &            CPD(30,30,25),UPD(30,30,25),VPD(30,30,25),
     &            WPD(30,30,25)
      COMMON/BL36/AP(30,30,25),AE(30,30,25),AW(30,30,25),AN(30,30,25),
     &            AS(30,30,25),AF(30,30,25),AB(30,30,25),SP(30,30,25),
     &            SU(30,30,25),RI(30,30,25)
      COMMON/BL37/VIS(30,30,25),COND(30,30,25),RESORM(40),
     &            CPM(30,30,25),NHSZ(3,2),NOD(30,30,25)
      COMMON/BL39/ALEW,CONSRA,QSIN,QSWER,QSWAL,QSAIR,QSFAN,QSHOL
      COMMON/BL50/JHWALS,JHWALF,KHWALS,KHWALF,NVENT

C *** CALCULATE COEFFICIENTS
      DO 100 K=2,NK+3
      DO 100 J=2,NJ+3
      DO 100 I=2,NI+3
```

```
C *** CENTRAL LENGTH OF THE SCALAR CONTROL VOLUME
          DXP1=DXXC(I+1)
          DXI =DXXC(I)
          DXM1=DXXC(I-1)

          DYP1=DYYC(J+1)
          DYJ =DYYC(J)
          DYM1=DYYC(J-1)

          DZP1=DZZC(K+1)
          DZK =DZZC(K)
          DZM1=DZZC(K-1)

C *** SURFACE LENGTH OF THE CONTROL VOLUME
          DXN=DXXC(I)
          DXS=DXXC(I)
          DXF=DXXC(I)
          DXB=DXXC(I)

          DYF=DYYC(J)
          DYB=DYYC(J)
          DYE=DYYC(J)
          DYW=DYYC(J)

          DZE=DZZC(K)
          DZW=DZZC(K)
          DZN=DZZC(K)
          DZS=DZZC(K)

C *** CENTRAL LENGTH OF THE STAGGERED CONTROL VOLUME FOR T
          DXEE=DXXS(I+2)
          DXE =DXXS(I+1)
          DXW =DXXS(I)
          DXWW=DXXS(I-1)

          DYNN=DYYS(J+2)
          DYN =DYYS(J+1)
          DYS =DYYS(J)
          DYSS=DYYS(J-1)

          DZFF=DZZS(K+2)
          DZF =DZZS(K+1)
          DZB =DZZS(K)
          DZBB=DZZS(K-1)

C *** DEFINE THE AREA OF THE CONTROL VOLUME
          DXYF=DXF*DYF
          DXYB=DXB*DYB
          DYZE=DYE*DZE
          DYZW=DYW*DZW
          DZXN=DZN*DXN
          DZXS=DZS*DXS

          VOL=DXI*DYJ*DZK
          VOLDT=VOL/DTIME

          ZXOYN=DZXN/DYN
          ZXOYS=DZXS/DYS
          XYOZF=DXYF/DZF
          XYOZB=DXYB/DZB
```

```
      YZOXE=DYZE/DXE
      YZOXW=DYZW/DXW

C *** DENSITY AT THE SURFACES OF THE CONTROL VOLUME
      GN=(R(I,J,K)*DYP1+R(I   ,J+1,K  )*DYJ)/(DYP1+DYJ)
      GS=(R(I,J,K)*DYM1+R(I   ,J-1,K  )*DYJ)/(DYM1+DYJ)
      GE=(R(I,J,K)*DXP1+R(I+1,J   ,K  )*DXI)/(DXP1+DXI)
      GW=(R(I,J,K)*DXM1+R(I-1,J   ,K  )*DXI)/(DXM1+DXI)
      GF=(R(I,J,K)*DZP1+R(I   ,J   ,K+1)*DZK)/(DZP1+DZK)
      GB=(R(I,J,K)*DZM1+R(I   ,J   ,K-1)*DZK)/(DZM1+DZK)

      CN=GN*V(I  ,J+1,K  )*DZXN
      CS=GS*V(I  ,J  ,K  )*DZXS
      CE=GE*U(I+1,J  ,K  )*DYZE
      CW=GW*U(I  ,J  ,K  )*DYZW
      CF=GF*W(I  ,J  ,K+1)*DXYF
      CB=GB*W(I  ,J  ,K  )*DXYB

C *** DIFFUSIVITY AT THE SURFACES OF THE CONTROL VOLUME
      CONDN=(DYP1+DYJ)/(DYJ/COND(I,J,K)+DYP1/COND(I   ,J+1,K  ))
      CONDS=(DYM1+DYJ)/(DYJ/COND(I,J,K)+DYM1/COND(I   ,J-1,K  ))
      CONDE=(DXP1+DXI)/(DXI/COND(I,J,K)+DXP1/COND(I+1,J   ,K  ))
      CONDW=(DXM1+DXI)/(DXI/COND(I,J,K)+DXM1/COND(I-1,J   ,K  ))
      CONDF=(DZP1+DZK)/(DZK/COND(I,J,K)+DZP1/COND(I   ,J   ,K+1))
      CONDB=(DZM1+DZK)/(DZK/COND(I,J,K)+DZM1/COND(I   ,J   ,K-1))

      CONDN1=ZXOYN*CONDN*ALEW
      CONDS1=ZXOYS*CONDS*ALEW
      CONDE1=YZOXE*CONDE*ALEW
      CONDW1=YZOXW*CONDW*ALEW
      CONDF1=XYOZF*CONDF*ALEW
      CONDB1=XYOZB*CONDB*ALEW

C *** QUICK SCHEME
      CEP=(ABS(CE)+CE)*DXP1*DXI/(DXE*(DXE+DXW ))/8.
      CEM=(ABS(CE)-CE)*DXP1*DXI/(DXE*(DXE+DXEE))/8.
      CWP=(ABS(CW)+CW)*DXM1*DXI/(DXW*(DXW+DXWW))/8.
      CWM=(ABS(CW)-CW)*DXM1*DXI/(DXW*(DXW+DXE ))/8.

      CNP=(ABS(CN)+CN)*DYP1*DYJ/(DYN*(DYN+DYS ))/8.
      CNM=(ABS(CN)-CN)*DYP1*DYJ/(DYN*(DYN+DYNN))/8.
      CSP=(ABS(CS)+CS)*DYM1*DYJ/(DYS*(DYS+DYSS))/8.
      CSM=(ABS(CS)-CS)*DYM1*DYJ/(DYS*(DYS+DYN ))/8.

      CFP=(ABS(CF)+CF)*DZP1*DZK/(DZF*(DZF+DZB ))/8.
      CFM=(ABS(CF)-CF)*DZP1*DZK/(DZF*(DZF+DZFF))/8.
      CBP=(ABS(CB)+CB)*DZM1*DZK/(DZB*(DZB+DZBB))/8.
      CBM=(ABS(CB)-CB)*DZM1*DZK/(DZB*(DZB+DZF ))/8.

      AE(I,J,K)=-.5*CE*DXI/DXE+CEP+CEM*(1.+DXE/DXEE)+CWM*DXW/DXE
      AW(I,J,K)= .5*CW*DXI/DXW+CWM+CWP*(1.+DXW/DXWW)+CEP*DXE/DXW
      AN(I,J,K)=-.5*CN*DYJ/DYN+CNP+CNM*(1.+DYN/DYNN)+CSM*DYS/DYN
      AS(I,J,K)= .5*CS*DYJ/DYS+CSM+CSP*(1.+DYS/DYSS)+CNP*DYN/DYS
      AF(I,J,K)=-.5*CF*DZK/DZF+CFP+CFM*(1.+DZF/DZFF)+CBM*DZB/DZF
      AB(I,J,K)= .5*CB*DZK/DZB+CBM+CBP*(1.+DZB/DZBB)+CFP*DZF/DZB

C *** BOUNDARY CONSIDERATION
      IF (I.LT.NI+3) THEN
         AEE=-CEM*DXE/DXEE
         AEER=AEE*CPD(I+2,J,K)
      ELSE
```

```
            AEE=0.
            AEER=0.
         ENDIF

         IF (I.GT.2) THEN
            AWW=-CWP*DXW/DXWW
            AWWR=AWW*CPD(I-2,J,K)
         ELSE
            AWW=0.
            AWWR=0.
         ENDIF

         IF (J.LT.NJ+3) THEN
            ANN=-CNM*DYN/DYNN
            ANNR=ANN*CPD(I,J+2,K)
         ELSE
            ANN=0.
            ANNR=0.
         ENDIF

         IF (J.GT.2) THEN
            ASS=-CSP*DYS/DYSS
            ASSR=ASS*CPD(I,J-2,K)
         ELSE
            ASS=0.
            ASSR=0.
         ENDIF

         IF (K.LT.NK+3) THEN
            AFF=-CFM*DZF/DZFF
            AFFR=AFF*CPD(I,J,K+2)
         ELSE
            AFF=0.
            AFFR=0.
         ENDIF

         IF (K.GT.2) THEN
            ABB=-CBP*DZB/DZBB
            ABBR=ABB*CPD(I,J,K-2)
         ELSE
            ABB=0.
            ABBR=0.
         ENDIF

C *** MODIFICATION FOR DECK BOUNDARIES
         IF (NOD(I-1,J,K).NE.0) THEN
            AWW=0.0
            AWWR=0.0
         ENDIF

         IF (NOD(I+1,J,K).NE.0) THEN
            AEE=0.0
            AEER=0.0
         ENDIF

         IF (NOD(I,J-1,K).NE.0) THEN
            ASS=0.0
            ASSR=0.0
         ENDIF

         IF (NOD(I,J+1,K).NE.0) THEN
```

```
            ANN=0.0
            ANNR=0.0
         ENDIF

         IF (NOD(I,J,K-1).NE.0) THEN
            ABB=0.0
            ABBR=0.0
         ENDIF

         IF (NOD(I,J,K+1).NE.0) THEN
            AFF=0.0
            AFFR=0.0
         ENDIF

         AP(I,J,K)=AE(I,J,K)+AW(I,J,K)+AN(I,J,K)+AS(I,J,K)+AF(I,J,K)+
     &              AB(I,J,K)+AEE+AWW+ANN+ASS+AFF+ABB+CONDE1+CONDW1+
     &              CONDN1+CONDS1+CONDF1+CONDB1

         AE(I,J,K)=AE(I,J,K)+CONDE1
         AW(I,J,K)=AW(I,J,K)+CONDW1
         AN(I,J,K)=AN(I,J,K)+CONDN1
         AS(I,J,K)=AS(I,J,K)+CONDS1
         AF(I,J,K)=AF(I,J,K)+CONDF1
         AB(I,J,K)=AB(I,J,K)+CONDB1

         SP(I,J,K)=-ROD(I,J,K)*VOLDT
         SU(I,J,K)=-SP(I,J,K)*COD(I,J,K)+AEER+AWWR+ANNR+ASSR+AFFR+ABBR
  100 CONTINUE

C *** TAKE CARE OF B.C. THRU AN,AS,AE,AW,AF,AB,SP AND SU

C *** Y DIRECTION
      DO 500 I=2,NI+3
      DO 500 K=2,NK+3
         SP(I,3   ,K)=SP(I,3   ,K)+AS(I,3   ,K)
         SP(I,NJ+2,K)=SP(I,NJ+2,K)+AN(I,NJ+2,K)
         AS(I,3   ,K)=0.
         AN(I,NJ+2,K)=0.
  500 CONTINUE

C *** X DIRECTION
      DO 600 J=2,NJ+3
      DO 600 K=2,NK+3
         SP(3   ,J,K)=SP(3   ,J,K)+AW(3   ,J,K)
         SP(NI+2,J,K)=SP(NI+2,J,K)+AE(NI+2,J,K)
         AW(3   ,J,K)=0.0
         AE(NI+2,J,K)=0.0
  600 CONTINUE

C *** Z DIRECTION
      DO 700 I=2,NI+3
      DO 700 J=2,NJ+3
         SP(I,J,3   )=SP(I,J,3   )+AB(I,J,3   )
         SP(I,J,NK+2)=SP(I,J,NK+2)+AF(I,J,NK+2)
         AB(I,J,3   )=0.
         AF(I,J,NK+2)=0.
 700  CONTINUE

C *** ASSEMBLE COEFFICIENTS AND SOLVE DIFFERENCE EQUATIONS
      DO 300 K=2,NK+3
      DO 300 J=2,NJ+3
```

```
      DO 300 I=2,NI+3
         AP(I,J,K)=AP(I,J,K)-SP(I,J,K)
  300 CONTINUE

C *** VOLUMETRIC MASS SOURCE INPUT
      VOLT=0.0
      DO 113 I=2,NI+3
      DO 113 J=2,NJ+3
      DO 113 K=2,NK+3
         DXI =DXXC(I)
         DYJ =DYYC(J)
         DZK =DZZC(K)
         VOL =DXI*DYJ*DZK*H**3
         VOLT=VOLT+VOL
  113 CONTINUE

      DO 111 I=NHSZ(1,1),NHSZ(1,2)
      DO 111 J=NHSZ(2,1),NHSZ(2,2)
      DO 111 K=NHSZ(3,1),NHSZ(3,2)
         DXI =DXXC(I)
         DYJ =DYYC(J)
         DZK =DZZC(K)
         VOL =DXI*DYJ*DZK
         SU(I,J,K)=SU(I,J,K)+VOL*H/(U0*RHO0*VOLT)
  111 CONTINUE

C *** SOLVE FOR C
      CALL TRID (3,3,3,NI+2,NJ+2,NK+2,C)

C *** Z DIRECTION
      DO 74 I=1,NI+4
      DO 74 J=1,NJ+4
         C(I,J,2   )=C(I,J,3   )
         C(I,J,1   )=C(I,J,2   )
         C(I,J,NK+3)=C(I,J,NK+2)
         C(I,J,NK+4)=C(I,J,NK+3)
   74 CONTINUE

C *** Y DIRECTION
      DO 84 I=2,NI+3
      DO 84 K=2,NK+3
         C(I,NJ+3,K)=C(I,NJ+4,K)
         C(I,NJ+4,K)=C(I,NJ+3,K)
         C(I,2   ,K)=C(I,   3,K)
         C(I,1   ,K)=C(I,   2,K)
  84  CONTINUE

C *** X DIRECTION
      DO 80 J=1,NJ+4
      DO 80 K=1,NK+4
         C(2   ,J,K)=C(   3,J,K)
         C(1   ,J,K)=C(   2,J,K)
         C(NI+3,J,K)=C(NI+2,J,K)
         C(NI+4,J,K)=C(NI+3,J,K)
  80  CONTINUE

      RETURN
      END

****************************************************************************
****************************************************************************
```

```
      SUBROUTINE CALP
******************************************************************************
*CALCULATES NODE PRESSURES

      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      COMMON/R4/XC(40),YC(40),ZC(40),XS(40),YS(40),ZS(40),DXXC(40),
     &          DYYC(40),DZZC(40),DXXS(40),DYYS(40),DZZS(40)
      COMMON/BL1/DX,DY,DZ,DTIME,TCOOL,PI,Q,QR
      COMMON/BL7/NI,NJ,NK,KRUN,NBLOR,NWRP
      COMMON/BL12/NWRITE,NTAPE,NTMAX0,NTREAL,TIME,SORSUM,ITER
      COMMON/BL22/CPS(20),CONS(20),WFAN(20),NCHIP,ICHPB(20),NCHPI(20),
     &            JCHPB(20),NCHPJ(20),KCHPB(20),NCHPK(20)
      COMMON/BL23/RMS(20),NMS,IMSB(20),NMSI(20),JMSB(20),NMSJ(20),
     &            KMSB(20),NMSK(20)
      COMMON/BL31/TOD(30,30,25),ROD(30,30,25),POD(30,30,25),
     &            COD(30,30,25),UOD(30,30,25),VOD(30,30,25),
     &            WOD(30,30,25)
      COMMON/BL32/T(30,30,25),R(30,30,25),P(30,30,25),C(30,30,25),
     &            U(30,30,25),V(30,30,25),W(30,30,25)
      COMMON/BL34/HEIGHT(30,30,25),REQ(30,30,25),SMP(30,30,25),
     &            SMPP(30,30,25),PP(30,30,25),DU(30,30,25),
     &            DV(30,30,25),DW(30,30,25)
      COMMON/BL36/AP(30,30,25),AE(30,30,25),AW(30,30,25),AN(30,30,25),
     &            AS(30,30,25),AF(30,30,25),AB(30,30,25),SP(30,30,25),
     &            SU(30,30,25),RI(30,30,25)
      COMMON/BL37/VIS(30,30,25),COND(30,30,25),RESORM(40),
     &            CPM(30,30,25),NHSZ(3,2),NOD(30,30,25)
      COMMON/BL50/JHWALS,JHWALF,KHWALS,KHWALF,NVENT
      COMMON/BL52/CONVMT

C *** CALCULATE COEFFICIENTS
      DO 100 K=2,NK+3
      DO 100 J=2,NJ+3
      DO 100 I=2,NI+3

      IF (NOD(I,J,K).EQ.1) GOTO 100

C *** CENTRAL LENGTH OF THE SCALAR CONTROL VOLUME
         DXP1=DXXC(I+1)
         DXI =DXXC(I)
         DXM1=DXXC(I-1)

         DYP1=DYYC(J+1)
         DYJ =DYYC(J)
         DYM1=DYYC(J-1)

         DZP1=DZZC(K+1)
         DZK =DZZC(K)
         DZM1=DZZC(K-1)

C *** SURFACE LENGTH OF THE CONTROL VOLUME
         DXN=DXXC(I)
         DXS=DXXC(I)
         DXF=DXXC(I)
         DXB=DXXC(I)

         DYF=DYYC(J)
         DYB=DYYC(J)
         DYE=DYYC(J)
         DYW=DYYC(J)
```

```
      DZE=DZZC(K)
      DZW=DZZC(K)
      DZN=DZZC(K)
      DZS=DZZC(K)

C *** DEFINE AREA OF THE CONTROL VOLUME
      DXYF=DXF*DYF
      DXYB=DXB*DYB
      DYZE=DYE*DZE
      DYZW=DYW*DZW
      DZXN=DZN*DXN
      DZXS=DZS*DXS

      VOL=DXI*DYJ*DZK
      VOLDT=VOL/DTIME

C *** DENSITY AT THE SURFACES
      RN=(R(I,J,K)*DYP1+R(I   ,J+1,K  )*DYJ)/(DYP1+DYJ)
      RS=(R(I,J,K)*DYM1+R(I   ,J-1,K  )*DYJ)/(DYM1+DYJ)
      RE=(R(I,J,K)*DXP1+R(I+1,J   ,K  )*DXI)/(DXP1+DXI)
      RW=(R(I,J,K)*DXM1+R(I-1,J   ,K  )*DXI)/(DXM1+DXI)
      RF=(R(I,J,K)*DZP1+R(I   ,J   ,K+1)*DZK)/(DZP1+DZK)
      RB=(R(I,J,K)*DZM1+R(I   ,J   ,K-1)*DZK)/(DZM1+DZK)

      AN(I,J,K)=RN*DZXN*DV(I   ,J+1,K  )
      AS(I,J,K)=RS*DZXS*DV(I   ,J   ,K  )
      AE(I,J,K)=RE*DYZE*DU(I+1,J   ,K  )
      AW(I,J,K)=RW*DYZW*DU(I   ,J   ,K  )
      AF(I,J,K)=RF*DXYF*DW(I   ,J   ,K+1)
      AB(I,J,K)=RB*DXYB*DW(I   ,J   ,K  )

      CN=RN*V(I   ,J+1,K  )*DZXN
      CS=RS*V(I   ,J   ,K  )*DZXS
      CE=RE*U(I+1,J   ,K  )*DYZE
      CW=RW*U(I   ,J   ,K  )*DYZW
      CF=RF*W(I   ,J   ,K+1)*DXYF
      CB=RB*W(I   ,J   ,K  )*DXYB

      SMP(I,J,K)=-(R(I,J,K)-ROD(I,J,K))*VOLDT-CE+CW-CN+CS-CF+CB
      SU (I,J,K)= SMP(I,J,K)
      SP (I,J,K)= 0.
  100 CONTINUE

C *** CONSIDER THE MASS SOURCE INPUT INTO THE CONTROL VOLUME
      IF (NMS.GE.1) THEN
         DO 150 M=1,NMS
            IB=IMSB(M)
            IE=IB+NMSI(M)-1
            JB=JMSB(M)
            JE=JB+NMSJ(M)-1
            KB=KMSB(M)
            KE=KB+NMSK(M)-1
            DO 160 I=IB,IE-1
            DO 160 J=JB,JE-1
            DO 160 K=KB,KE-1
               SU(I,J,K)=SU(I,J,K)+RMS(M)
  160       CONTINUE
  150    CONTINUE
      ENDIF

C *** TAKE CARE OF B.C. THRU AN,AS,AE,AW,AF,AB,SP AND SU
```

```
C *** Y DIRECTION
      DO 500 K=2,NK+3
      DO 500 I=2,NI+3
         AS(I,2   ,K)=0.
         AN(I,NJ+3,K)=0.
  500 CONTINUE
      IF (NVENT.LT.0) GOTO 498
      DO 499 K=KHWALS,KHWALF
      DO 499 J=JHWALS,JHWALF
          SP(2,J,K)=SP(2,J,K)-AW(2,J,K)
 499  CONTINUE
 498  CONTINUE
C *** X DIRECTION
      DO 501 K=2,NK+3
      DO 501 J=2,NJ+3
       IF(NVENT.GT.0) THEN
         IF((K.LE.KHWALS-1 .OR. K.GE.KHWALF+1).OR.
     &     (J.LE.JHWALS-1 .OR. J.GE.JHWALF+1)) THEN
            AW(2   ,J,K)=0.
         END IF
       ELSE
         AW(2   ,J,K)=0.
       END IF
         AE(NI+3,J,K)=0.
  501 CONTINUE

C *** Z -DIRECTION
      DO 502 I=2,NI+3
      DO 502 J=2,NJ+3
         AB(I,J,2   )=0.
         AF(I,J,NK+3)=0.
 502  CONTINUE

C *** MODIFICATION FOR DECK BOUNDARIES
      IF (NCHIP.EQ.0) GOTO 110
      DO 101 N=1,NCHIP
         IB  =ICHPB(N)
         IE  =IB+NCHPI(N)-1
         JB  =JCHPB(N)
         JE  =JB+NCHPJ(N)-1
         KB  =KCHPB(N)
         KE  =KB+NCHPK(N)-1

         DO 102 J=JB,JE-1
         DO 102 K=KB,KE-1
            AE(IB-1,J,K)=0.0
            AW(IE  ,J,K)=0.0
  102    CONTINUE

         DO 103 I=IB,IE-1
         DO 103 K=KB,KE-1
            AN(I,JB-1,K)=0.0
            AS(I,JE  ,K)=0.0
  103    CONTINUE

         DO 106 I=IB,IE-1
         DO 106 J=JB,JE-1
            AF(I,J,KB-1)=0.0
            AB(I,J,KE  )=0.0
  106    CONTINUE
```

```
C *** FOR THE CELLS INSIDE OF THE DECKS
          DO 104 I=IB,IE-1
          DO 104 J=JB,JE-1
          DO 104 K=KB,KE-1
             SP(I,J,K)=-1.0E2
             AW(I,J,K)=0.
             AE(I,J,K)=0.
             AS(I,J,K)=0.
             AN(I,J,K)=0.
             AB(I,J,K)=0.
             AF(I,J,K)=0.
             SU(I,J,K)=0.
  104     CONTINUE
  101 CONTINUE

C *** ASSEMBLE COEFFICIENTS AND SOLVE DIFFERENCE EQUATIONS
  110 DO 300 I=2,NI+3
      DO 300 J=2,NJ+3
      DO 300 K=2,NK+3
         AP(I,J,K)=AN(I,J,K)+AS(I,J,K)+AE(I,J,K)+AW(I,J,K)-SP(I,J,K)
     &             +AF(I,J,K)+AB(I,J,K)
  300 CONTINUE
      IF(NVENT.LT.0) GOTO 598
      DO 599 K=KHWALS,KHWALF
      DO 599 J=JHWALS,JHWALF
        AW(2,J,K)=0.0
 599  CONTINUE
 598  CONTINUE
C *** SOLUTION OF FINITE DIFFERENCE EQUATION
      CALL TRID  (3,3,3,NI+2,NJ+2,NK+2,PP)

C *** CORRECTION FOR VELOCITY U
      DO 600 I=3,NI+3
      DO 600 J=2,NJ+3
      DO 600 K=2,NK+3
         U(I,J,K)=U(I,J,K)+DU(I,J,K)*(PP(I-1,J,K)-PP(I,J,K))
  600 CONTINUE

C *** CORRECTION FOR VELOCITY V
      DO 603 J=3,NJ+3
      DO 603 K=2,NK+3
      DO 603 I=2,NI+3
         V(I,J,K)=V(I,J,K)+DV(I,J,K)*(PP(I,J-1,K)-PP(I,J,K))
  603 CONTINUE

C *** CORRECTION FOR VELOCITY W
      DO 604 K=3,NK+3
      DO 604 I=2,NI+3
      DO 604 J=2,NJ+3
         W(I,J,K)=W(I,J,K)+DW(I,J,K)*(PP(I,J,K-1)-PP(I,J,K))
 604  CONTINUE

C *** CORRECTION FOR PRESSURE P
      DO 606 J=1,NJ+4
      DO 606 I=1,NI+4
      DO 606 K=1,NK+4
         P (I,J,K)=P(I,J,K)+PP(I,J,K)
         PP(I,J,K)=0.
  606 CONTINUE
```

```
C *** RESET THE VELOCITY INSIDE OF DECK
      IF (NCHIP.EQ.0) GOTO 121
      DO 120 N=1,NCHIP
         IB=ICHPB(N)
         IE=IB+NCHPI(N)-1
         JB=JCHPB(N)
         JE=JB+NCHPJ(N)-1
         KB=KCHPB(N)
         KE=KB+NCHPK(N)-1
         DO 109 I=IB,IE
         DO 109 J=JB,JE-1
         DO 109 K=KB,KE-1
            U(I,J,K)=0.0
  109    CONTINUE

         DO 118 I=IB,IE-1
         DO 118 J=JB,JE
         DO 118 K=KB,KE-1
            V(I,J,K)=0.0
  118    CONTINUE

         DO 119 I=IB,IE-1
         DO 119 J=JB,JE-1
         DO 119 K=KB,KE
            W(I,J,K)=WFAN(N)
  119    CONTINUE
  120 CONTINUE

C *** RECALCULATE THE ERROR SOURCE AFTER CORRECTIONS OF U, V, P
  121 SORSUM=0.
      RESORM(ITER)=0.
      DO 700 J=2,NJ+3
      DO 700 I=2,NI+3
      DO 700 K=2,NK+3
         IF (NOD(I,J,K).NE.1) THEN

C *** CENTRAL LENGTH OF THE SCALAR CONTROL VOLUME
            DXP1=DXXC(I+1)
            DXI =DXXC(I)
            DXM1=DXXC(I-1)

            DYP1=DYYC(J+1)
            DYJ =DYYC(J)
            DYM1=DYYC(J-1)

            DZP1=DZZC(K+1)
            DZK =DZZC(K)
            DZM1=DZZC(K-1)

C *** SURFACE LENGTH OF THE CONTROL VOLUME
            DXN=DXXC(I)
            DXS=DXXC(I)
            DXF=DXXC(I)
            DXB=DXXC(I)

            DYF=DYYC(J)
            DYB=DYYC(J)
            DYE=DYYC(J)
            DYW=DYYC(J)
```

```
            DZE=DZZC(K)
            DZW=DZZC(K)
            DZN=DZZC(K)
            DZS=DZZC(K)

C *** DEFINE AREA OF THE CONTROL VOLUME
            DXYF=DXF*DYF
            DXYB=DXB*DYB
            DYZE=DYE*DZE
            DYZW=DYW*DZW
            DZXN=DZN*DXN
            DZXS=DZS*DXS

            VOL=DXI*DYJ*DZK
            VOLDT=VOL/DTIME

C *** CALCULATE DENSITY
            RN=(R(I,J,K)*DYP1+R(I  ,J+1,K  )*DYJ)/(DYP1+DYJ)
            RS=(R(I,J,K)*DYM1+R(I  ,J-1,K  )*DYJ)/(DYM1+DYJ)
            RE=(R(I,J,K)*DXP1+R(I+1,J  ,K  )*DXI)/(DXP1+DXI)
            RW=(R(I,J,K)*DXM1+R(I-1,J  ,K  )*DXI)/(DXM1+DXI)
            RF=(R(I,J,K)*DZP1+R(I  ,J  ,K+1)*DZK)/(DZP1+DZK)
            RB=(R(I,J,K)*DZM1+R(I  ,J  ,K-1)*DZK)/(DZM1+DZK)

            CN=RN*V(I  ,J+1,K  )*DZXN
            CS=RS*V(I  ,J  ,K  )*DZXS
            CE=RE*U(I+1,J  ,K  )*DYZE
            CW=RW*U(I  ,J  ,K  )*DYZW
            CF=RF*W(I  ,J  ,K+1)*DXYF
            CB=RB*W(I  ,J  ,K  )*DXYB
            SMP(I,J,K)=(ROD(I,J,K)-R(I,J,K))*VOLDT-CE+CW-CN+CS-CF+CB

C *** SORSUM IS ACTUAL MASS INCREASE OR DECREASE FROM CONTINUITY
C *** EQUATION, THIS WILL BE COMPARED TO MASS SOURCE

C *** CONSIDER THE MASS SOURCE INPUT INTO THE CONTROL VOLUME
            IF (NMS.GT.0) THEN
               DO 250 M=1,NMS
                  IB=IMSB(M)
                  IE=IB+NMSI(M)-1
                  JB=JMSB(M)
                  JE=JB+NMSJ(M)-1
                  KB=KMSB(M)
                  KE=KB+NMSK(M)-1
                  DO 260 II=IB,IE-1
                  DO 260 JJ=JB,JE-1
                  DO 260 KK=KB,KE-1
                     IF ((II.EQ.I).AND.(JJ.EQ.J).AND.(KK.EQ.K)) THEN
                        SMP(I,J,K)=SMP(I,J,K)+RMS(M)
                     ENDIF
  260             CONTINUE
  250          CONTINUE
            ENDIF
            SORSUM=SORSUM+SMP(I,J,K)

C *** RESORM IS SUM OF THE ABSOLUTE VALUE OF SMP(I,J,K)
            RESORM(ITER)=RESORM(ITER)+ABS(SMP(I,J,K))
         ENDIF

C *** APPROXIMATE MASS CONVECTED FROM COMPT THRU VENT
        IF (NVENT.LT.0) GOTO 700
```

```
        IF (I.EQ.3) THEN
         IF ((J.GE.JHWALS.AND.J.LE.JHWALF).AND.
     &       (K.GE.KHWALS.AND.K.LE.JHWALF)) THEN
           CONVMT=CONVMT+CW*DTIME
         END IF
        END IF

  700 CONTINUE
      RETURN
      END

************************************************************************
************************************************************************
      SUBROUTINE CALQ
************************************************************************
*
*VARIABLES:
*  BR     = MAXIMUM BURN RATE (LBM/SEC)
*  F      = MAXIMUM FUEL AVAILABLE (LBM)
*  FR     = TOTAL FUEL REMAINING (LBM)
*  H      = REFERENCE LENGTH (FT)
*  HC     = HEAT OF COMBUSTION (BTU/LBM)
*  HSTART= FIRE START TIME (SECONDS)
*  Q      = TOTAL HEAT INPUT (BTU/SEC)
*  TIME   = NONDIMENSIONAL FIRE TIME
*  U0     = REFERENCE VELOCITY (FT/SEC)
*  XTIME = FIRE TIME (SECONDS)
*
************************************************************************

      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      COMMON/BL1/DX,DY,DZ,DTIME,TCOOL,PI,Q,QR
      COMMON/BL2/X,Y,H,TFLR,TWAL
      COMMON/BL3/F,FR,HSTART
      COMMON/BL12/NWRITE,NTAPE,NTMAX0,NTREAL,TIME,SORSUM,ITER
      COMMON/BL16/U0,UGRT,BUOY,CP0,PRT,COND0,VIS0,RHO0,
     &            TA,DTEMP,TWRITE,TTAPE,TMAX,GC,RAIR,NT

      XTIME=TIME*H/U0
      HC=2600.0
      BR=  0.50

C *** CALCULATE HEAT RELEASE RATE (Q) IN BTU/SEC
C *** NOTE:  THESE ALGORITHMS ASSUME A LINEAR INCREASE IN BOTH
C ***        HEAT RELEASE AND FUEL CONSUMPTION OVER THE FIRST
C ***        INDICATED  NUMBER OF SECONDS OF FIRE TIME,
C ***        AFTER WHICH BOTH ARE AT MAXIMUM
      IF(XTIME.LT.HSTART) THEN
         Q=0.0
         FR=F
      ELSEIF(XTIME.GE.HSTART.AND.(XTIME-HSTART).LE.8.0) THEN
         IF(FR.LE.0.0) THEN
            Q =0.0
            FR=0.0
         ELSE
            Q =HC*BR*(XTIME-HSTART)/8.
            FR=F-BR*(XTIME-HSTART)**2/8.
         ENDIF
      ELSEIF((XTIME-HSTART).GT.8.0) THEN
         IF(FR.LE.0.0) THEN
            Q =0.0
```

```
            FR=0.0
         ELSE
            Q =HC*BR
            FR=F-BR*(XTIME-HSTART)
         ENDIF
      ENDIF

C *** TAKE RADIATION HEAT FLUX INTO ACCOUNT
      Q=Q-QR
      IF (Q.LE.0.0) Q=0.

      RETURN
      END

************************************************************************
************************************************************************
      SUBROUTINE CALT
************************************************************************

      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      COMMON/R4/XC(40),YC(40),ZC(40),XS(40),YS(40),ZS(40),DXXC(40),
     &          DYYC(40),DZZC(40),DXXS(40),DYYS(40),DZZS(40)
      COMMON/BL1/DX,DY,DZ,DTIME,TCOOL,PI,Q,QR
      COMMON/BL2/X,Y,H,TFLR,TWAL
      COMMON/BL7/NI,NJ,NK,KRUN,NBLOR,NWRP
      COMMON/BL14/HCOEF,CNT,ABTURB,BTURB,VISL,VISMAX
      COMMON/BL16/U0,UGRT,BUOY,CP0,PRT,COND0,VIS0,RHO0,
     &            TA,DTEMP,TWRITE,TTAPE,TMAX,GC,RAIR,NT
      COMMON/BL23/RMS(20),NMS,IMSB(20),NMSI(20),JMSB(20),NMSJ(20),
     &            KMSB(20),NMSK(20)
      COMMON/BL22/CPS(20),CONS(20),WFAN(20),NCHIP,ICHPB(20),NCHPI(20),
     &            JCHPB(20),NCHPJ(20),KCHPB(20),NCHPK(20)
      COMMON/BL31/TOD(30,30,25),ROD(30,30,25),POD(30,30,25),
     &            COD(30,30,25),UOD(30,30,25),VOD(30,30,25),
     &            WOD(30,30,25)
      COMMON/BL32/T(30,30,25),R(30,30,25),P(30,30,25),C(30,30,25),
     &            U(30,30,25),V(30,30,25),W(30,30,25)
      COMMON/BL33/TPD(30,30,25),RPD(30,30,25),PPD(30,30,25),
     &            CPD(30,30,25),UPD(30,30,25),VPD(30,30,25),
     &            WPD(30,30,25)
      COMMON/BL34/HEIGHT(30,30,25),REQ(30,30,25),SMP(30,30,25),
     &            SMPP(30,30,25),PP(30,30,25),DU(30,30,25),
     &            DV(30,30,25),DW(30,30,25)
      COMMON/BL36/AP(30,30,25),AE(30,30,25),AW(30,30,25),AN(30,30,25),
     &            AS(30,30,25),AF(30,30,25),AB(30,30,25),SP(30,30,25),
     &            SU(30,30,25),RI(30,30,25)
      COMMON/BL37/VIS(30,30,25),COND(30,30,25),RESORM(40),
     &            CPM(30,30,25),NHSZ(3,2),NOD(30,30,25)
      COMMON/BL39/ALEW,CONSRA,QSIN,QSWER,QSWAL,QSAIR,QSFAN,QSHOL
      COMMON/BL43/QSCONF,QSCONB,QSCONE,QSCONW,QSCONN,QSCONS,QSCONH,
     &            QSRADF, QSRADB,QSRADE,QSRADW,QSRADN,QSRADS,QSRHOL,
     &            WAIR,WWAL,WINS,WERR,WWFAN
      COMMON/BL50/JHWALS,JHWALF,KHWALS,KHWALF,NVENT
      COMMON/BL51/SSGPW(20,20,20)
      COMMON/BL52/CONVMT

      WRES=0.0

C *** NONDIMENSIONAL REFERENCE TEMPERATURE
      TINF=TA/TA
```

```
C *** CALCULATE COEFFICIENTS
      DO 100 K=2,NK+3
      DO 100 J=2,NJ+3
      DO 100 I=2,NI+3

C *** CENTRAL LENGTH OF THE TEMPERTURE CONTROL VOLUME
         DXP1=DXXC(I+1)
         DXI =DXXC(I)
         DXM1=DXXC(I-1)

         DYP1=DYYC(J+1)
         DYJ =DYYC(J)
         DYM1=DYYC(J-1)

         DZP1=DZZC(K+1)
         DZK =DZZC(K)
         DZM1=DZZC(K-1)

C *** SURFACE LENGTH OF THE CONTROL VOLUME
         DXN=DXXC(I)
         DXS=DXXC(I)
         DXF=DXXC(I)
         DXB=DXXC(I)

         DYF=DYYC(J)
         DYB=DYYC(J)
         DYE=DYYC(J)
         DYW=DYYC(J)

         DZE=DZZC(K)
         DZW=DZZC(K)
         DZN=DZZC(K)
         DZS=DZZC(K)

C *** CENTRAL LENGTH OF THE STAGGERED CONTROL VOLUME FOR TEMPERATURE
         DXEE=DXXS(I+2)
         DXE =DXXS(I+1)
         DXW =DXXS(I)
         DXWW=DXXS(I-1)

         DYNN=DYYS(J+2)
         DYN =DYYS(J+1)
         DYS =DYYS(J)
         DYSS=DYYS(J-1)

         DZFF=DZZS(K+2)
         DZF =DZZS(K+1)
         DZB =DZZS(K)
         DZBB=DZZS(K-1)

C *** DEFINE THE AREA OF THE CONTROL VOLUME
         DXYF=DXF*DYF
         DXYB=DXB*DYB
         DYZE=DYE*DZE
         DYZW=DYW*DZW
         DZXN=DZN*DXN
         DZXS=DZS*DXS

         VOL=DXI*DYJ*DZK
         VOLDT=VOL/DTIME
```

```
C *** FOR CONDUCTION
          ZXOYN=DZXN/DYN
          ZXOYS=DZXS/DYS
          XYOZF=DXYF/DZF
          XYOZB=DXYB/DZB
          YZOXE=DYZE/DXE
          YZOXW=DYZW/DXW

C *** DENSITY AT THE SURFACES
          GN=(R(I,J,K)*DYP1+R(I,J+1,K)*DYJ)/(DYP1+DYJ)
          GS=(R(I,J,K)*DYM1+R(I,J-1,K)*DYJ)/(DYM1+DYJ)
          GE=(R(I,J,K)*DXP1+R(I+1,J,K)*DXI)/(DXP1+DXI)
          GW=(R(I,J,K)*DXM1+R(I-1,J,K)*DXI)/(DXM1+DXI)
          GF=(R(I,J,K)*DZP1+R(I,J,K+1)*DZK)/(DZP1+DZK)
          GB=(R(I,J,K)*DZM1+R(I,J,K-1)*DZK)/(DZM1+DZK)

C *** THE MASS FLUX RATE THROUGH THE SURFACES
          CN=GN*V(I,J+1,K)*DZXN
          CS=GS*V(I,J  ,K)*DZXS
          CE=GE*U(I+1,J,K)*DYZE
          CW=GW*U(I  ,J,K)*DYZW
          CF=GF*W(I,J,K+1)*DXYF
          CB=GB*W(I,J,K  )*DXYB

C *** CONDUCTIVITY AT THE SURFACES
          CONDN=(DYP1+DYJ)*COND(I,J,K)*COND(I,J+1,K)*DYJ*DYP1/
     &           (DYJ*COND(I,J,K)+DYP1*COND(I,J+1,K))
          CONDS=(DYM1+DYJ)*COND(I,J,K)*COND(I,J-1,K)*DYJ*DYM1/
     &           (DYJ*COND(I,J,K)+DYM1*COND(I,J-1,K))
          CONDE=(DXP1+DXI)*COND(I,J,K)*COND(I+1,J,K)*DXI*DXP1/
     &           (DXI*COND(I,J,K)+DXP1*COND(I+1,J,K))
          CONDW=(DXM1+DXI)*COND(I,J,K)*COND(I-1,J,K)*DXI*DXM1/
     &           (DXI*COND(I,J,K)+DXM1*COND(I-1,J,K))
          CONDF=(DZP1+DZK)*COND(I,J,K)*COND(I,J,K+1)*DZK*DZP1/
     &           (DZK*COND(I,J,K)+DZP1*COND(I,J,K+1))
          CONDB=(DZM1+DZK)*COND(I,J,K)*COND(I,J,K-1)*DZK*DZM1/
     &           (DZK*COND(I,J,K)+DZM1*COND(I,J,K-1))

C *** CONDUCTION COMPONENT
          CONDN1=ZXOYN*CONDN
          CONDS1=ZXOYS*CONDS
          CONDE1=YZOXE*CONDE
          CONDW1=YZOXW*CONDW
          CONDF1=XYOZF*CONDF
          CONDB1=XYOZB*CONDB

C *** QUICK SCHEME
          CEP=(ABS(CE)+CE)*DXP1*DXI/(DXE*(DXE+DXW )*8.)
          CEM=(ABS(CE)-CE)*DXP1*DXI/(DXE*(DXE+DXEE)*8.)
          CWP=(ABS(CW)+CW)*DXM1*DXI/(DXW*(DXW+DXWW)*8.)
          CWM=(ABS(CW)-CW)*DXM1*DXI/(DXW*(DXW+DXE )*8.)

          CNP=(ABS(CN)+CN)*DYP1*DYJ/(DYN*(DYN+DYS )*8.)
          CNM=(ABS(CN)-CN)*DYP1*DYJ/(DYN*(DYN+DYNN)*8.)
          CSP=(ABS(CS)+CS)*DYM1*DYJ/(DYS*(DYS+DYSS)*8.)
          CSM=(ABS(CS)-CS)*DYM1*DYJ/(DYS*(DYS+DYN )*8.)

          CFP=(ABS(CF)+CF)*DZP1*DZK/(DZF*(DZF+DZB )*8.)
          CFM=(ABS(CF)-CF)*DZP1*DZK/(DZF*(DZF+DZFF)*8.)
          CBP=(ABS(CB)+CB)*DZM1*DZK/(DZB*(DZB+DZBB)*8.)
          CBM=(ABS(CB)-CB)*DZM1*DZK/(DZB*(DZB+DZF )*8.)
```

```
      AE(I,J,K)=-.5*CE*DXI/DXE+CEP+CEM*(1.+DXE/DXEE)+CWM*DXW/DXE
      AW(I,J,K)= .5*CW*DXI/DXW+CWM+CWP*(1.+DXW/DXWW)+CEP*DXE/DXW
      AN(I,J,K)=-.5*CN*DYJ/DYN+CNP+CNM*(1.+DYN/DYNN)+CSM*DYS/DYN
      AS(I,J,K)= .5*CS*DYJ/DYS+CSM+CSP*(1.+DYS/DYSS)+CNP*DYN/DYS
      AF(I,J,K)=-.5*CF*DZK/DZF+CFP+CFM*(1.+DZF/DZFF)+CBM*DZB/DZF
      AB(I,J,K)= .5*CB*DZK/DZB+CBM+CBP*(1.+DZB/DZBB)+CFP*DZF/DZB

C *** BOUNDARY CONSIDERATIONS
      IF (I.LT.NI+3) THEN
         AEE=-CEM*DXE/DXEE
         AEER=AEE*TPD(I+2,J,K)*CPM(I+2,J,K)
      ELSE
         AEE=0.
         AEER=0.
      ENDIF

      IF (I.GT.2) THEN
         AWW=-CWP*DXW/DXWW
         AWWR=AWW*TPD(I-2,J,K)*CPM(I-2,J,K)
      ELSE
         AWW=0.
         AWWR=0.
      ENDIF

      IF (J.LT.NJ+3) THEN
         ANN=-CNM*DYN/DYNN
         ANNR=ANN*TPD(I,J+2,K)*CPM(I,J+2,K)
      ELSE
         ANN=0.
         ANNR=0.
      ENDIF

      IF (J.GT.2) THEN
         ASS=-CSP*DYS/DYSS
         ASSR=ASS*TPD(I,J-2,K)*CPM(I,J-2,K)
      ELSE
         ASS=0.
         ASSR=0.
      ENDIF

      IF (K.LT.NK+3) THEN
         AFF=-CFM*DZF/DZFF
         AFFR=AFF*TPD(I,J,K+2)*CPM(I,J,K+2)
      ELSE
         AFF=0.
         AFFR=0.
      ENDIF

      IF (K.GT.2) THEN
         ABB=-CBP*DZB/DZBB
         ABBR=ABB*TPD(I,J,K-2)*CPM(I,J,K-2)
      ELSE
         ABB=0.
         ABBR=0.
      ENDIF

C *** MODIFICATION FOR DECK BOUNDARIES
      IF (NOD(I-1,J,K).NE.0) THEN
         AWW=0.0
         AWWR=0.0
```

```
         ENDIF

         IF (NOD(I+1,J,K).NE.0) THEN
            AEE=0.0
            AEER=0.0
         ENDIF

         IF (NOD(I,J-1,K).NE.0) THEN
            ASS=0.0
            ASSR=0.0
         ENDIF

         IF (NOD(I,J+1,K).NE.0) THEN
            ANN=0.0
            ANNR=0.0
         ENDIF

         IF (NOD(I,J,K-1).NE.0) THEN
            ABB=0.0
            ABBR=0.0
         ENDIF

         IF (NOD(I,J,K+1).NE.0) THEN
            AFF=0.0
            AFFR=0.0
         ENDIF

         AP(I,J,K)=(AE(I,J,K)+AW(I,J,K)+AN(I,J,K)+AS(I,J,K)+AF(I,J,K)+
     &               AB(I,J,K)+AEE+AWW+ANN+ASS+AFF+ABB)*CPM(I,J,K)+
     &               CONDE1+CONDW1+CONDN1+CONDS1+CONDF1+CONDB1
         AE(I,J,K)=AE(I,J,K)*CPM(I+1,J  ,K  )+CONDE1
         AW(I,J,K)=AW(I,J,K)*CPM(I-1,J  ,K  )+CONDW1
         AN(I,J,K)=AN(I,J,K)*CPM(I  ,J+1,K  )+CONDN1
         AS(I,J,K)=AS(I,J,K)*CPM(I  ,J-1,K  )+CONDS1
         AF(I,J,K)=AF(I,J,K)*CPM(I  ,J  ,K+1)+CONDF1
         AB(I,J,K)=AB(I,J,K)*CPM(I  ,J  ,K-1)+CONDB1

         SP(I,J,K)=-ROD(I,J,K)*VOLDT*CPM(I,J,K)
         SU(I,J,K)=-SP(I,J,K)*TOD(I,J,K)+AEER+AWWR+ANNR+ASSR+AFFR+ABBR
  100 CONTINUE

C *** TAKE CARE OF B.C. THRU AN,AS,AE,AW,AF,AB,SP AND SU

C *** Y-DIRECTION
      DO 500 I=3,NI+2
      DO 500 K=3,NK+2
         SU(I,3   ,K)=SU(I,3   ,K)+AS(I,3   ,K)*T(I,2   ,K)
         SU(I,NJ+2,K)=SU(I,NJ+2,K)+AN(I,NJ+2,K)*T(I,NJ+3,K)
         AS(I,3   ,K)=0.
         AN(I,NJ+2,K)=0.
  500 CONTINUE

C *** X-DIRECTION
      DO 600 J=3,NJ+2
      DO 600 K=3,NK+2
       IF(NVENT.GT.0) THEN
        IF((J.LE.JHWALS-1 .OR. J.GE.JHWALF+1).OR.
     &      (K.LE.KHWALS-1 .OR. K.GE.KHWALF+1)) THEN
         SU(3   ,J,K)=SU(3   ,J,K)+AW(3   ,J,K)*T(2   ,J,K)
          AW(3   ,J,K)=0.0
        END IF
```

```
        ELSE
          SU(3    ,J,K)=SU(3    ,J,K)+AW(3    ,J,K)*T(2    ,J,K)
          AW(3    ,J,K)=0.0
        END IF
          SU(NI+2,J,K)=SU(NI+2,J,K)+AE(NI+2,J,K)*T(NI+3,J,K)
          AE(NI+2,J,K)=0.0
  600 CONTINUE

      IF(NVENT.LT.0) GOTO 598
      DO 610 J=JHWALS,JHWALF
      DO 610 K=KHWALS,KHWALF
       IF(U(3,J,K).LT.0.0) THEN
          T(2,J,K)= T(3,J,K)
       ELSE
         T(2,J,K)=1.0
       END IF
         SU(3    ,J,K)=SU(3    ,J,K)+AW(3    ,J,K)*T(2,J,K)
         AW(3,J,K)=0.0
 610  CONTINUE
 598  CONTINUE
C *** Z-DIRECTION
      DO 700 I=3,NI+2
      DO 700 J=3,NJ+2
      SU(I,J,3    )=SU(I,J,3    )+AB(I,J,3    )*T(I,J,2    )
      SU(I,J,NK+2)=SU(I,J,NK+2)+AF(I,J,NK+2)*T(I,J,NK+3)
      AB(I,J,3    )=0.
      AF(I,J,NK+2)=0.
 700  CONTINUE

C *** CONSIDER THE MASS SOURCE INPUT TO THE CONTROL VOLUME
      IF (NMS.GE.1) THEN
         DO 150 M=1,NMS
            IB=IMSB(M)
            IE=IB+NMSI(M)-1
            JB=JMSB(M)
            JE=JB+NMSJ(M)-1
            KB=KMSB(M)
            KE=KB+NMSK(M)-1
            DO 160 I=IB,IE-1
            DO 160 J=JB,JE-1
            DO 160 K=KB,KE-1
               IF (RMS(M).GE.0.0) THEN
                  RMSCPT=RMS(M)*1.0*CPM(I,J,K)
               ELSE
                  RMSCPT=RMS(M)*T(I,J,K)*CPM(I,J,K)*R(I,J,K)
               ENDIF
  160       CONTINUE
  150    CONTINUE
      ENDIF

C *** CONSIDER THE RADIATION HEAT FLUX FROM THE FIRE TO THE BLOCK
      CALL RADHT (2)

C *** ASSEMBLE COEFFICIENTS AND SOLVE DIFFERENCE EQUATIONS
      DO 300 K=3,NK+2
      DO 300 J=3,NJ+2
      DO 300 I=3,NI+2
         AP(I,J,K)=AP(I,J,K)-SP(I,J,K)
  300 CONTINUE

C *** VOLUME HEAT SOURCE INPUT
```

```
C *** CALCULATE THE TOTAL VOLUME OCCUPIED BY HEAT SOURCE
C *** DISTRIBUTE ENERGY INTO EACH CONTROL VOLUME
C *** QQQ/H**3    DIMENSIONLESS HEAT SOURCE

      QQQ=Q*H/(U0*CP0*RHO0*TA)
      VOLT=0.0
      DO 113 I=NHSZ(1,1),NHSZ(1,2)
      DO 113 J=NHSZ(2,1),NHSZ(2,2)
      DO 113 K=NHSZ(3,1),NHSZ(3,2)
         VOL=DXXC(I)*DYYC(J)*DZZC(K)
         VOLT=VOLT+VOL*H**3
  113 CONTINUE
      DO 114 I=NHSZ(1,1),NHSZ(1,2)
      DO 114 J=NHSZ(2,1),NHSZ(2,2)
      DO 114 K=NHSZ(3,1),NHSZ(3,2)
         VOL=DXXC(I)*DYYC(J)*DZZC(K)
         SU(I,J,K)=SU(I,J,K)+VOL*QQQ/VOLT
  114 CONTINUE

C *** SOLVE FOR T
      CALL TRID (3,3,3,NI+2,NJ+2,NK+2,T)

      DO 2001 I=1,NI+4
      DO 2001 J=1,NJ+4
      DO 2001 K=1,NK+4
         IF(T(I,J,K).LT.TCOOL) T(I,J,K)=TCOOL
 2001 CONTINUE
C *** CALCULATE RADIATION HEAT TRANSFER

C *** HERE SU(I,J,K) IS USED TO STORE THE RADIATIVE HEAT FLUX
      DO 75 I=1,NI+4
      DO 75 J=1,NJ+4
      DO 75 K=1,NK+4
         SU(I,J,K)=0.
  75  CONTINUE

C *** CONSIDER THE RADIATION HEAT FLUX FROM THE FIRE TO THE WALL
      CALL RADHT (1)

C *** SUMMATION OF CONDUCTION HEAT FLUX AND RADIATION HEAT FLUX TO WALLS
      QSCONF=0.
      QSCONB=0.
      QSCONE=0.
      QSCONW=0.
      QSCONN=0.
      QSCONS=0.
      QSCONH=0.

      QSRADF=0.
      QSRADB=0.
      QSRADE=0.
      QSRADW=0.
      QSRADN=0.
      QSRADS=0.
      QSRHOL=0.


C *** CALCULATE CONDUCTION, RADIATION & TEMPERATURE ON THE SOLID WALLS
      DO 74 I=3,NI+2
```

```
      DO 74 J=3,NJ+2

C *** ON THE BACK WALL
         DZK =DZZC(2)
         DZP1=DZZC(3)
         DXI =DXXC(I)
         DYJ =DYYC(J)
         DXY =DXI*DYJ
         VOL =DXY*DZK
         CONDF=(DZP1+DZK)*DZK*DZP1*COND(I,J,2)*COND(I,J,3)/
     &          (DZK*COND(I,J,2)+DZP1*COND(I,J,3))
         QCONF=DXY*CONDF*(T(I,J,3)-T(I,J,2))*2.0/(DZP1+DZK)
         QCONB=DXY*COND(I,J,2)*(T(I,J,1)-T(I,J,2))*2.0/DZK
         QRADB=SU(I,J,2)

         T(I,J,2)=TOD(I,J,2)+DTIME*(QCONF+QCONB+QRADB)/(VOL*CPM(I,J,2))
         T(I,J,1)=(2.*COND(I,J,2)*T(I,J,2)+HCOEF*TINF*DZK)/
     &             (HCOEF*DZK+2.*COND(I,J,2))

         QSCONB=QSCONB+QCONF
         QSRADB=QSRADB+QRADB

C *** ON THE FRONT WALL
         DZK =DZZC(NK+3)
         DZM1=DZZC(NK+2)
         DXI =DXXC(I)
         DYJ =DYYC(J)
         DXY =DXI*DYJ
         VOL =DXY*DZK
         CONDB=(DZM1+DZK)*DZK*DZM1*COND(I,J,NK+3)*COND(I,J,NK+2)/
     &          (DZK*COND(I,J,NK+3)+DZM1*COND(I,J,NK+2))

         QCONB=DXY*CONDB*(T(I,J,NK+2)-T(I,J,NK+3))*2.0/(DZK+DZM1)
         QCONF=DXY*COND(I,J,NK+3)*(T(I,J,NK+4)-T(I,J,NK+3))*2.0/DZK
         QRADF=SU(I,J,NK+3)

         T(I,J,NK+3)=TOD(I,J,NK+3)+DTIME*(QCONB+QCONF+QRADF)/
     &                (VOL*CPM(I,J,NK+3))
         T(I,J,NK+4)=(2.0*COND(I,J,NK+3)*T(I,J,NK+3)+HCOEF*TINF*DZK)/
     &                (HCOEF*DZK+2.0*COND(I,J,NK+3))

         QSCONF=QSCONF+QCONB
         QSRADF=QSRADF+QRADF

   74 CONTINUE

      DO 84 I=3,NI+2
      DO 84 K=3,NK+2

C *** ON THE SOUTH WALL
         DYJ =DYYC(2)
         DYP1=DYYC(3)
         DXI =DXXC(I)
         DZK =DZZC(K)
         DZX =DZK*DXI
         VOL =DZX*DYJ
         CONDN=(DYP1+DYJ)*DYJ*DYP1*COND(I,2,K)*COND(I,3,K)/
     &          (DYJ*COND(I,2,K)+DYP1*COND(I,3,K))

         QCONN=DZX*CONDN*(T(I,3,K)-T(I,2,K))*2.0/(DYP1+DYJ)
         QCONS=DZX*COND(I,2,K)*(T(I,1,K)-T(I,2,K))*2.0/DYJ
```

```
      QRADS=SU(I,2,K)
      T(I,2,K)=TOD(I,2,K)+DTIME*(QCONN+QCONS+QRADS)/
     &             (VOL*CPM(I,2,K))
      T(I,1,K)=(2.0*COND(I,2 ,K)*T(I,2 ,K)+HCOEF*TINF*DYJ)/
     &             (HCOEF*DYJ+2.0*COND(I,2 ,K))

      QSCONS=QSCONS+QCONN
      QSRADS=QSRADS+QRADS

C *** ON THE NORTH WALL
      DYJ =DYYC(NJ+3)
      DYM1=DYYC(NJ+2)
      DXI =DXXC(I)
      DZK =DZZC(K)
      DZX =DZK*DXI
      VOL =DZX*DYJ
      CONDS=(DYM1+DYJ)*DYJ*DYM1*COND(I,NJ+3,K)*COND(I,NJ+2,K)/
     &         (DYJ*COND(I,NJ+3,K)+DYM1*COND(I,NJ+2,K))

      QCONS=DZX*CONDS*(T(I,NJ+2,K)-T(I,NJ+3,K))*2.0/(DYM1+DYJ)
      QCONN=DZX*COND(I,NJ+3,K)*(T(I,NJ+4,K)-T(I,NJ+3,K))*2.0/DYJ
      QRADN=SU(I,NJ+3,K)
      T(I,NJ+3,K)=TOD(I,NJ+3,K)+DTIME*(QCONS+QCONN+QRADN)/
     &                 (VOL*CPM(I,NJ+3,K))
      T(I,NJ+4,K)=(2.0*COND(I,NJ+3,K)*T(I,NJ+3,K)+HCOEF*TINF*DYJ)/
     &                 (HCOEF*DYJ+2.0*COND(I,NJ+3,K))

      QSCONN=QSCONN+QCONS
      QSRADN=QSRADN+QRADN
   84 CONTINUE

      DO 80 J=3,NJ+2
      DO 80 K=3,NK+2

C *** ON THE EAST WALL
      DXI =DXXC(NI+3)
      DXM1=DXXC(NI+2)
      DYJ =DYYC(J)
      DZK =DZZC(K)
      DYZ =DYJ*DZK
      VOL =DYZ*DXI
      CONDW=(DXM1+DXI)*DXI*DXM1*COND(NI+3,J,K)*COND(NI+2,J,K)/
     &         (DXI*COND(NI+3,J,K)+DXM1*COND(NI+2,J,K))


      QCONW=DYZ*CONDW       *(T(NI+2,J,K)-T(NI+3,J,K))*2.0/(DXI+DXM1)
      QCONE=DYZ*COND(2,J,K)*(T(NI+4,J,K)-T(NI+3,J,K))*2.0/DXI
      QRADE=SU(NI+3,J,K)
      T(NI+3,J,K)=TOD(NI+3,J,K)+DTIME*(QCONW+QCONE+QRADE)/
     &                 (VOL*CPM(NI+3,J,K))
      T(NI+4,J,K)=(2.0*COND(NI+3,J,K)*T(NI+3,J,K)+HCOEF*TINF*DXI)/
     &                 (HCOEF*DXI+2.0*COND(NI+3,J,K))

      QSCONE=QSCONE+QCONW
      QSRADE=QSRADE+QRADE
   80 CONTINUE

      DO 92 J=3,NJ+3
      DO 92 K=3,NK+3

       IF(NVENT.GT.0) THEN
```

```
         IF ((J.LE.JHWALS-1 .OR. J.GE.JHWALF+1).OR.
     &     (K.LE.KHWALS-1 .OR. K.GE.KHWALF+1)) THEN
C *** ON THE WEST WALL
          DXI =DXXC(2)
          DXP1=DXXC(3)
          DYJ =DYYC(J)
          DZK =DZZC(K)
          DYZ =DYJ*DZK
          VOL =DYZ*DXI
          CONDE=(DXP1+DXI)*DXI*DXP1*COND(2,J,K)*COND(3,J,K)/
     &          (DXI*COND(2,J,K)+DXP1*COND(3,J,K))
          QCONE=DYZ*CONDE *(T(3,J,K)-T(2,J,K))*2.0/(DXI+DXP1)
          QCONW=DYZ*COND(2,J,K)*(T(1,J,K)
     &                                        -T(2,J,K))*2.0/DXI
          QRADW=SU(2,J,K)
          T(2,J,K)=TOD(2,J,K)+DTIME
     &             *(QCONE+QCONW+QRADW)/(VOL*CPM(2,J,K))
          T(1,J,K)=(2.0*COND(2,J,K)*T(2,J,K)
     &             +HCOEF*TINF*DXI)/
     &             (HCOEF*DXI+2.0*COND(2,J,K))

          QSCONW=QSCONW+QCONE
          QSRADW=QSRADW+QRADW
         END IF

        ELSE

          DXI =DXXC(2)
          DXP1=DXXC(3)
          DYJ =DYYC(J)
          DZK =DZZC(K)
          DYZ =DYJ*DZK
          VOL =DYZ*DXI
          CONDE=(DXP1+DXI)*DXI*DXP1*COND(2,J,K)*COND(3,J,K)/
     &          (DXI*COND(2,J,K)+DXP1*COND(3,J,K))

          QCONE=DYZ*CONDE *(T(3,J,K)-T(2,J,K))*2.0/(DXI+DXP1)
          QCONW=DYZ*COND(2,J,K)*(T(1,J,K)
     &                                        -T(2,J,K))*2.0/DXI

          QRADW=SU(2,J,K)
          T(2,J,K)=TOD(2,J,K)+DTIME
     &             *(QCONE+QCONW+QRADW)/(VOL*CPM(2,J,K))
          T(1,J,K)=(2.0*COND(2,J,K)*T(2,J,K)
     &             +HCOEF*TINF*DXI)/
     &             (HCOEF*DXI+2.0*COND(2,J,K))

          QSCONW=QSCONW+QCONE
          QSRADW=QSRADW+QRADW
        END IF

 92   CONTINUE


      IF(NVENT.GT.0) THEN
       DXW=DXXC(2)
       DXE=DXXC(3)
       DO 96 J=JHWALS,JHWALF
       DO 96 K=KHWALS,KHWALF
         DYJ=DYYC(J)
```

```
         DZK=DZZC(K)
         DYZ=DYJ*DZK
         GW=SILIN(R(2,J,K),R(3,J,K),DYW,DXE)*U(3,J,K)
         TW=SILIN(T(2,J,K),T(3,J,K),DYW,DXE)
         QSCONH=QSCONH+TW*GW*CPM(3,J,K)*DYZ
         QSRHOL=QSRHOL+SSGPW(2,J,K)
         SSGPW(2,J,K)=0.0
 96     CONTINUE
      ELSE
         QSCONH=0.0
         QSRHOL=0.0
      END IF

C *** CALCULATE THE ENERGY LOST THROUGH (OR CONSUMED BY)
C     1) THE CAVITY  WALLS; 2) CAVITY  AIR; 3) DUCT

C *** TANK  AIR
      WERR=0.
      WAIR=0.
      DO 25 I=3,NI+2
      DO 25 J=3,NJ+2
      DO 25 K=3,NK+2
         IF(NOD(I,J,K).EQ.1) GO TO 25
         DXI=DXXC(I)
         DYJ=DYYC(J)
         DZK=DZZC(K)

C *** CENTRAL LENGTH OF THE SCALAR CONTROL VOLUME
         DXP1=DXXC(I+1)
         DXI =DXXC(I)
         DXM1=DXXC(I-1)

         DYP1=DYYC(J+1)
         DYJ =DYYC(J)
         DYM1=DYYC(J-1)

         DZP1=DZZC(K+1)
         DZK =DZZC(K)
         DZM1=DZZC(K-1)

C *** SURFACE LENGTH OF THE CONTROL VOLUME
         DXN=DXXC(I)
         DXS=DXXC(I)
         DXF=DXXC(I)
         DXB=DXXC(I)

         DYF=DYYC(J)
         DYB=DYYC(J)
         DYE=DYYC(J)
         DYW=DYYC(J)

         DZE=DZZC(K)
         DZW=DZZC(K)
         DZN=DZZC(K)
         DZS=DZZC(K)

C *** DEFINE AREA OF THE CONTROL VOLUME
         DXYF=DXF*DYF
         DXYB=DXB*DYB
         DYZE=DYE*DZE
         DYZW=DYW*DZW
```

```
      DZXN=DZN*DXN
      DZXS=DZS*DXS

      VOL=DXI*DYJ*DZK
      VOLDT=VOL/DTIME

      RN=(R(I,J,K)*DYP1+R(I,J+1,K)*DYJ)/(DYP1+DYJ)
      RS=(R(I,J,K)*DYM1+R(I,J-1,K)*DYJ)/(DYM1+DYJ)
      RE=(R(I,J,K)*DXP1+R(I+1,J,K)*DXI)/(DXP1+DXI)
      RW=(R(I,J,K)*DXM1+R(I-1,J,K)*DXI)/(DXM1+DXI)
      RF=(R(I,J,K)*DZP1+R(I,J,K+1)*DZK)/(DZP1+DZK)
      RB=(R(I,J,K)*DZM1+R(I,J,K-1)*DZK)/(DZM1+DZK)

      CN=RN*V(I,J+1,K)*DZXN
      CS=RS*V(I,J  ,K)*DZXS
      CE=RE*U(I+1,J,K)*DYZE
      CW=RW*U(I  ,J,K)*DYZW
      CF=RF*W(I,J,K+1)*DXYF
      CB=RB*W(I,J,K  )*DXYB

      WERR=WERR+T(I,J,K)*CPM(I,J,K)*SMP(I,J,K)
      WRES=WRES+SMP(I,J,K)
      WAIR=WAIR+(T(I,J,K)*R(I,J,K)-TOD(I,J,K)*ROD(I,J,K))*
   &        CPM(I,J,K)*VOLDT
 25 CONTINUE

C *** SUM UP THE TOTAL WALT AT THAT TIME STEP (DIMENSIONLESS)
      WINS=QQQ/H**3

C *** SUM UP THE TOTAL WALT LOST TO THE WALLS
      QTCON=QSCONF+QSCONB+QSCONS+QSCONN+QSCONW+QSCONE
      QTRAD=QSRADF+QSRADB+QSRADS+QSRADN+QSRADW+QSRADE

      WWAL=QTCON+QTRAD
      WWHOL=QSRHOL+QSCONH

C *** EQUIVALENT INTERNAL HEAT SOURCE DUE TO RADIATION
      QR=QTRAD*U0*CP0*RHO0*TA*H**2

C *** TOTAL WALT EXHAUSTED THROUGH THE DUCT
      IF (NMS.EQ.0) THEN
         WWFAN=0.0
      ELSE
         WWFAN=8000.*CPM(12,3,12)*(T(12,3,12)-1.0)*R(12,3,12)/
     &        (60.*H**2*U0)
      ENDIF

C *** THE ENERGY CALCULATION
      QSIN=QSIN+WINS*DTIME
      QSWER=QSWER+WERR*DTIME
      QSWAL=QSWAL+WWAL*DTIME
      QSAIR=QSAIR+WAIR*DTIME
      QSFAN=QSFAN+WWFAN*DTIME
      QSHOL=QSHOL+WWHOL*DTIME

      RETURN
      END

************************************************************************
************************************************************************
      SUBROUTINE CALU
```

```
************************************************************************
*CALCULATES THE U COMPONENT OF THE VELOCITY

       IMPLICIT DOUBLE PRECISION (A-H,O-Z)
       COMMON/R4/XC(40),YC(40),ZC(40),XS(40),YS(40),ZS(40),DXXC(40),
     &          DYYC(40),DZZC(40),DXXS(40),DYYS(40),DZZS(40)
       COMMON/BL1/DX,DY,DZ,DTIME,TCOOL,PI,Q,QR
       COMMON/BL7/NI,NJ,NK,KRUN,NBLOR,NWRP
       COMMON/BL20/SIG11(30,30,25),SIG12(30,30,25),SIG22(30,30,25),
     &            SIG13(30,30,25),SIG23(30,30,25),SIG33(30,30,25)
       COMMON/BL22/CPS(20),CONS(20),WFAN(20),NCHIP,ICHPB(20),NCHPI(20),
     &            JCHPB(20),NCHPJ(20),KCHPB(20),NCHPK(20)
       COMMON/BL31/TOD(30,30,25),ROD(30,30,25),POD(30,30,25),
     &            COD(30,30,25),UOD(30,30,25),VOD(30,30,25),
     &            WOD(30,30,25)
       COMMON/BL32/T(30,30,25),R(30,30,25),P(30,30,25),C(30,30,25),
     &            U(30,30,25),V(30,30,25),W(30,30,25)
       COMMON/BL33/TPD(30,30,25),RPD(30,30,25),PPD(30,30,25),
     &            CPD(30,30,25),UPD(30,30,25),VPD(30,30,25),
     &            WPD(30,30,25)
       COMMON/BL34/HEIGHT(30,30,25),REQ(30,30,25),SMP(30,30,25),
     &            SMPP(30,30,25),PP(30,30,25),DU(30,30,25),
     &            DV(30,30,25),DW(30,30,25)
       COMMON/BL36/AP(30,30,25),AE(30,30,25),AW(30,30,25),AN(30,30,25),
     &            AS(30,30,25),AF(30,30,25),AB(30,30,25),SP(30,30,25),
     &            SU(30,30,25),RI(30,30,25)
       COMMON/BL37/VIS(30,30,25),COND(30,30,25),RESORM(40),
     &            CPM(30,30,25),NHSZ(3,2),NOD(30,30,25)
       COMMON/BL50/JHWALS,JHWALF,KHWALS,KHWALF, NVENT

C *** CALCULATE COEFFICIENTS
      DO 100 K=2,NK+3
      DO 100 J=2,NJ+3
      DO 100 I=3,NI+3

C *** CENTRAL LENGTH OF THE U CONTROL VOLUME
         DXP1=DXXS(I+1)
         DXI =DXXS(I)
         DXM1=DXXS(I-1)

         DYP1=DYYC(J+1)
         DYJ =DYYC(J)
         DYM1=DYYC(J-1)

         DZP1=DZZC(K+1)
         DZK =DZZC(K)
         DZM1=DZZC(K-1)

C *** SURFACE LENGTH OF THE CONTROL VOLUME
         DXN=DXXS(I)
         DXS=DXXS(I)
         DXF=DXXS(I)
         DXB=DXXS(I)

         DYF=DYYC(J)
         DYB=DYYC(J)
         DYE=DYYC(J)
         DYW=DYYC(J)

         DZE=DZZC(K)
         DZW=DZZC(K)
```

```
      DZN=DZZC(K)
      DZS=DZZC(K)

C *** CENTRAL LENGTH OF THE STAGGERED CONTROL VOLUME FOR U
      DXEE=DXXC(I+1)
      DXE =DXXC(I)
      DXW =DXXC(I-1)
      DXWW=DXXC(I-2)

      DYNN=DYYS(J+2)
      DYN =DYYS(J+1)
      DYS =DYYS(J)
      DYSS=DYYS(J-1)

      DZFF=DZZS(K+2)
      DZF =DZZS(K+1)
      DZB =DZZS(K)
      DZBB=DZZS(K-1)

C *** DEFINE THE AREA OF THE CONTROL VOLUME
      DXYF=DXF*DYF
      DXYB=DXB*DYB
      DYZE=DYE*DZE
      DYZW=DYW*DZW
      DZXN=DZN*DXN
      DZXS=DZS*DXS

      VOL=DXI*DYJ*DZK
      VOLDT=VOL/DTIME

      ZXOYN=DZXN/DYN
      ZXOYS=DZXS/DYS
      XYOZF=DXYF/DZF
      XYOZB=DXYB/DZB
      YZOXE=DYZE/DXE
      YZOXW=DYZW/DXW

C *** USE SINGLE AND BI-LINEAR INTERPOLATION TO EVALUATE
C     PHYSICAL PROPERTIES AND FLUX ON THE SURFACES.
      GNE=SILIN(R(I  ,J+1,K),R(I  ,J,K),DYP1,DYJ)*V(I  ,J+1,K)
      GNW=SILIN(R(I-1,J+1,K),R(I-1,J,K),DYP1,DYJ)*V(I-1,J+1,K)
      GSE=SILIN(R(I  ,J-1,K),R(I  ,J,K),DYM1,DYJ)*V(I  ,J  ,K)
      GSW=SILIN(R(I-1,J-1,K),R(I-1,J,K),DYM1,DYJ)*V(I-1,J  ,K)

      GE =SILIN(R(I+1,J,K),R(I  ,J,K),DXEE,DXE)*U(I+1,J,K)
      GP =SILIN(R(I-1,J,K),R(I  ,J,K),DXW ,DXE)*U(I  ,J,K)
      GW =SILIN(R(I-2,J,K),R(I-1,J,K),DXWW,DXW)*U(I-1,J,K)

      GFE=SILIN(R(I  ,J,K+1),R(I  ,J,K),DZP1,DZK)*W(I  ,J,K+1)
      GFW=SILIN(R(I-1,J,K+1),R(I-1,J,K),DZP1,DZK)*W(I-1,J,K+1)
      GBE=SILIN(R(I  ,J,K-1),R(I  ,J,K),DZM1,DZK)*W(I  ,J,K  )
      GBW=SILIN(R(I-1,J,K-1),R(I-1,J,K),DZM1,DZK)*W(I-1,J,K  )

C *** MASS FLOW RATE
      CE=0.5*(GE+GP)*DYZE
      CW=0.5*(GP+GW)*DYZW

      CN=SILIN(GNE,GNW,DXE,DXW)*DZXN
      CS=SILIN(GSE,GSW,DXE,DXW)*DZXS

      CF=SILIN(GFE,GFW,DXE,DXW)*DXYF
```

```
      CB=SILIN(GBE,GBW,DXE,DXW)*DXYB

C *** VISCOSITY
      VISE=VIS(I  ,J,K)
      VISW=VIS(I-1,J,K)

      VISN=(VIS(I,J+1,K)+VIS(I,J,K)+VIS(I-1,J+1,K)+VIS(I-1,J,K))/4.0
      VISS=(VIS(I,J-1,K)+VIS(I,J,K)+VIS(I-1,J-1,K)+VIS(I-1,J,K))/4.0

      VISF=(VIS(I,J,K+1)+VIS(I,J,K)+VIS(I-1,J,K+1)+VIS(I-1,J,K))/4.0
      VISB=(VIS(I,J,K-1)+VIS(I,J,K)+VIS(I-1,J,K-1)+VIS(I-1,J,K))/4.0

      VISN1=ZXOYN*VISN
      VISS1=ZXOYS*VISS
      VISE1=YZOXE*VISE
      VISW1=YZOXW*VISW
      VISF1=XYOZF*VISF
      VISB1=XYOZB*VISB

C *** QUICK SCHEME
      CEP=(ABS(CE)+CE)*DXE/(DXI *16.)
      CEM=(ABS(CE)-CE)*DXE/(DXP1*16.)
      CWP=(ABS(CW)+CW)*DXW/(DXM1*16.)
      CWM=(ABS(CW)-CW)*DXW/(DXI *16.)

      CNP=(ABS(CN)+CN)*DYP1*DYJ/(8.*DYN*(DYN+DYS ))
      CNM=(ABS(CN)-CN)*DYP1*DYJ/(8.*DYN*(DYN+DYNN))
      CSP=(ABS(CS)+CS)*DYM1*DYJ/(8.*DYS*(DYS+DYSS))
      CSM=(ABS(CS)-CS)*DYM1*DYJ/(8.*DYS*(DYS+DYN ))

      CFP=(ABS(CF)+CF)*DZP1*DZK/(8.*DZF*(DZF+DZB ))
      CFM=(ABS(CF)-CF)*DZP1*DZK/(8.*DZF*(DZF+DZFF))
      CBP=(ABS(CB)+CB)*DZM1*DZK/(8.*DZB*(DZB+DZBB))
      CBM=(ABS(CB)-CB)*DZM1*DZK/(8.*DZB*(DZB+DZF ))

      AE(I,J,K)= -.5*CE     +CWM*DXW /DXE+CEP+CEM*(1.+DXE/DXEE)+VISE1
      AW(I,J,K)=  .5*CW     +CEP*DXE /DXW+CWM+CWP*(1.+DXW/DXWW)+VISW1
      AN(I,J,K)=(-.5*CN*DYJ+CSM*DYS)/DYN+CNP+CNM*(1.+DYN/DYNN)+VISN1
      AS(I,J,K)=( .5*CS*DYJ+CNP*DYN)/DYS+CSM+CSP*(1.+DYS/DYSS)+VISS1
      AF(I,J,K)=(-.5*CF*DZK+CBM*DZB)/DZF+CFP+CFM*(1.+DZF/DZFF)+VISF1
      AB(I,J,K)=( .5*CB*DZK+CFP*DZF)/DZB+CBM+CBP*(1.+DZB/DZBB)+VISB1

C *** BOUNDARY CONSIDERATION
      IF(I.LT.NI+3) THEN
         AEE=-CEM*DXE/DXEE
         AEER=AEE*UPD(I+2,J,K)
      ELSE
         AEE=0.
         AEER=0.
      ENDIF

      IF (I.GT.3) THEN
         AWW=-CWP*DXW/DXWW
         AWWR=AWW*UPD(I-2,J,K)
      ELSE
         AWW=0.
         AWWR=0.
      ENDIF

      IF (J.LT.NJ+3) THEN
         ANN=-CNM*DYN/DYNN
```

```
         ANNR=ANN*UPD(I,J+2,K)
      ELSE
         ANN=0.
         ANNR=0.
      ENDIF

      IF (J.GT.2) THEN
         ASS=-CSP*DYS/DYSS
         ASSR=ASS*UPD(I,J-2,K)
      ELSE
         ASS=0.
         ASSR=0.
      ENDIF

      IF (K.LT.NK+3) THEN
         AFF=-CFM*DZF/DZFF
         AFFR=AFF*UPD(I,J,K+2)
      ELSE
         AFF=0.
         AFFR=0.
      ENDIF

      IF (K.GT.2) THEN
         ABB=-CBP*DZB/DZBB
         ABBR=ABB*UPD(I,J,K-2)
      ELSE
         ABB=0.
         ABBR=0.
      ENDIF

C *** MODIFICATION FOR DECK BOUNDARIES
      IF (NOD(I-2,J,K).NE.0) THEN
         AWW=0.0
         AWWR=0.0
      ENDIF

      IF (NOD(I+1,J,K).NE.0) THEN
         AEE=0.0
         AEER=0.0
      ENDIF

      IF (NOD(I,J-1,K).NE.0) THEN
         ASS=0.0
         ASSR=0.0
      ENDIF

      IF (NOD(I,J+1,K).NE.0) THEN
         ANN=0.0
         ANNR=0.0
      ENDIF

      IF (NOD(I,J,K-1).NE.0) THEN
         ABB=0.0
         ABBR=0.0
      ENDIF

      IF (NOD(I,J,K+1).NE.0) THEN
         AFF=0.0
         AFFR=0.0
      ENDIF
```

```
C *** SU FROM NORMAL STRESS
          RE=(SIG11(I   ,J,K)-(U(I+1,J,K)-U(I   ,J,K))*VISE/DXE)*DYZE
          RW=(SIG11(I-1,J,K)-(U(I   ,J,K)-U(I-1,J,K))*VISW/DXW)*DYZW

          RN=(SIG12(I,J+1,K)-(U(I,J+1,K)-U(I,J   ,K))*VISN/DYN)*DZXN
          RS=(SIG12(I,J   ,K)-(U(I,J   ,K)-U(I,J-1,K))*VISS/DYS)*DZXS

          RF=(SIG13(I,J,K+1)-(U(I,J,K+1)-U(I,J,K   ))*VISF/DZF)*DXYF
          RB=(SIG13(I,J,K   )-(U(I,J,K   )-U(I,J,K-1))*VISB/DZB)*DXYB

C *** SU FROM CURVED STRESSES AND ACCELERATIONS
          AVG12=0.5*(SIG12(I,J+1,K   )+SIG12(I,J,K))
          AVG13=0.5*(SIG13(I,J   ,K+1)+SIG13(I,J,K))

          AVG22=SILIN(SIG22(I,J,K),SIG22(I-1,J,K),DXE,DXW)
          AVG33=SILIN(SIG33(I,J,K),SIG33(I-1,J,K),DXE,DXW)

          AU1=U(I,J,K)
          AU2=BILIN(V(I   ,J+1,K),V(I   ,J,K),DYJ,DYJ,
     &              V(I-1,J+1,K),V(I-1,J,K),DYJ,DYJ,DXE,DXW)
          AU3=BILIN(W(I   ,J,K+1),W(I   ,J,K),DZK,DZK,
     &              W(I-1,J,K+1),W(I-1,J,K),DZK,DZK,DXE,DXW)

          AR=SILIN(R(I,J,K),R(I-1,J,K),DXE,DXW)

          ARU12=AR*AU1*AU2
          ARU13=AR*AU1*AU3
          ARU22=AR*AU2*AU2
          ARU33=AR*AU3*AU3

          RRY=(AVG12-ARU12)*DZK*(DXN-DXS)
          RRZ=(AVG13-ARU13)*DYJ*(DXF-DXB)
          RRX=(AVG22-ARU22)*DZK*(DYE-DYW)+(AVG33-ARU33)*DYJ*(DZE-DZW)

          AP(I,J,K)=AE(I,J,K)+AW(I,J,K)+AN(I,J,K)+AS(I,J,K)+AF(I,J,K)+
     &              AB(I,J,K)+AEE+AWW+ANN+ASS+AFF+ABB
          SP(I,J,K)=-(ROD(I,J,K)*DXW+ROD(I-1,J,K)*DXE)*VOLDT/(DXW+DXE)
          SU(I,J,K)=-SP(I,J,K)*UOD(I,J,K)+DYJ*DZK*(P(I-1,J,K)-P(I,J,K))+
     &              AEER+AWWR+ANNR+ASSR+AFFR+ABBR+RE-RW+RN-RS+RF-RB+RRY+
     &              RRZ-RRX
  100 CONTINUE

C *** TAKE CARE OF B.C. THRU AN,AS,AE,AW,AF,AB,SP AND SU

C *** Y DIRECTION
      DO 500 K=2,NK+3
      DO 500 I=3,NI+3
          SP(I,2   ,K)=SP(I,2   ,K)-AS(I,2   ,K)
          SP(I,NJ+3,K)=SP(I,NJ+3,K)-AN(I,NJ+3,K)
          AN(I,NJ+3,K)=0.
          AS(I,2   ,K)=0.
  500 CONTINUE

C *** X DIRECTION
      DO 502 K=2,NK+3
      DO 502 J=2,NJ+3
      IF(NVENT.GT.0) THEN
       IF((J.LE.JHWALS-1 .OR. J.GE.JHWALF+1).OR.
     &   (K.LE.KHWALS-1 .OR. K.GE.KHWALF+1)) THEN
         AW(3   ,J,K)=0.0
       END IF
```

```
      ELSE
         AW(3    ,J,K)=0.0
      END IF
         AE(NI+3,J,K)=0.0
  502 CONTINUE
      IF(NVENT.LT.0) GOTO 598
      DO 503 J=JHWALS,JHWALF
      DO 503 K=KHWALS,KHWALF
         SP(3,J  ,K)=SP(3,J    ,K)+AW(3,J    ,K)
         AW(3,J  ,K)=0.
  503 CONTINUE
  598 CONTINUE
C *** Z DIRECTION
      DO 600 I=3,NI+3
      DO 600 J=2,NJ+3
         SP(I,J,2    )=SP(I,J,2    )-AB(I,J,2    )
         SP(I,J,NK+3)=SP(I,J,NK+3)-AF(I,J,NK+3)
         AF(I,J,NK+3)=0.
         AB(I,J,2    )=0.
 600  CONTINUE

C *** MODIFICATION FOR DECK BOUNDARIES
      IF (NCHIP.EQ.0) GOTO 201
      DO 101 N=1,NCHIP
         IB  =ICHPB(N)
         IE  =IB+NCHPI(N)-1
         JB  =JCHPB(N)
         JE  =JB+NCHPJ(N)-1
         KB  =KCHPB(N)
         KE  =KB+NCHPK(N)-1

         DO 102 J=JB,JE-1
         DO 102 K=KB,KE-1
            AE(IB-1,J,K)=0.0
            AW(IE+1,J,K)=0.0
  102    CONTINUE

         DO 103 I=IB,IE
         DO 103 K=KB,KE-1
            SP(I,JB-1,K)=SP(I,JB-1,K)-AN(I,JB-1,K)
            AN(I,JB-1,K)=0.0
            SP(I,JE  ,K)=SP(I,JE  ,K)-AS(I,JE  ,K)
            AS(I,JE  ,K)=0.0
  103    CONTINUE

         DO 106 I=IB,IE
         DO 106 J=JB,JE-1
            SP(I,J,KB-1)=SP(I,J,KB-1)-AF(I,J,KB-1)
            AF(I,J,KB-1)=0.0
            SP(I,J,KE  )=SP(I,J,KE  )-AB(I,J,KE  )
            AB(I,J,KE  )=0.0
  106    CONTINUE

C *** FOR THE CELLS INSIDE OF THE DECKS
         DO 104 I=IB,IE
         DO 104 J=JB,JE-1
         DO 104 K=KB,KE-1
            SP(I,J,K)=-1.0E2
            AW(I,J,K)=0.
            AE(I,J,K)=0.
            AS(I,J,K)=0.
```

```
            AN(I,J,K)=0.
            AB(I,J,K)=0.
            AF(I,J,K)=0.
            SU(I,J,K)=0.
  104     CONTINUE
  101 CONTINUE

C *** ASSEMBLE COEFFICIENTS AND SOLVE DIFFERENCE EQUATIONS
  201 DO 301 K=2,NK+3
      DO 301 J=2,NJ+3
      DO 301 I=3,NI+3
         DYJ=DYYC(J)
         DZK=DZZC(K)
         DYZ=DYJ*DZK
         AP(I,J,K)=AP(I,J,K)-SP(I,J,K)
         DU(I,J,K)=DYZ/AP(I,J,K)
  301 CONTINUE

C *** SOLVE FOR U
      CALL TRID (4,3,3,NI+2,NJ+2,NK+2,U)


C *** RESET THE VELOCITY INSIDE OF DECK
      IF (NCHIP.EQ.0) GOTO 111
      DO 110 N=1,NCHIP
         IB=ICHPB(N)
         IE=IB+NCHPI(N)-1
         JB=JCHPB(N)
         JE=JB+NCHPJ(N)-1
         KB=KCHPB(N)
         KE=KB+NCHPK(N)-1
         DO 108 I=IB,IE
         DO 108 J=JB,JE-1
         DO 108 K=KB,KE-1
            U(I,J,K)=0.0
  108    CONTINUE
  110 CONTINUE

  111 RETURN
      END

************************************************************************
************************************************************************
      SUBROUTINE CALV
************************************************************************
*CALCULATES THE V COMPONENT OF THE VELOCITY

      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      COMMON/R4/XC(40),YC(40),ZC(40),XS(40),YS(40),ZS(40),DXXC(40),
     &          DYYC(40),DZZC(40),DXXS(40),DYYS(40),DZZS(40)
      COMMON/BL1/DX,DY,DZ,DTIME,TCOOL,PI,Q,QR
      COMMON/BL7/NI,NJ,NK,KRUN,NBLOR,NWRP
      COMMON/BL20/SIG11(30,30,25),SIG12(30,30,25),SIG22(30,30,25),
     &            SIG13(30,30,25),SIG23(30,30,25),SIG33(30,30,25)
      COMMON/BL22/CPS(20),CONS(20),WFAN(20),NCHIP,ICHPB(20),NCHPI(20),
     &            JCHPB(20),NCHPJ(20),KCHPB(20),NCHPK(20)
      COMMON/BL31/TOD(30,30,25),ROD(30,30,25),POD(30,30,25),
     &            COD(30,30,25),UOD(30,30,25),VOD(30,30,25),
     &            WOD(30,30,25)
      COMMON/BL32/T(30,30,25),R(30,30,25),P(30,30,25),C(30,30,25),
     &            U(30,30,25),V(30,30,25),W(30,30,25)
```

```
      COMMON/BL33/TPD(30,30,25),RPD(30,30,25),PPD(30,30,25),
     &            CPD(30,30,25),UPD(30,30,25),VPD(30,30,25),
     &            WPD(30,30,25)
      COMMON/BL34/HEIGHT(30,30,25),REQ(30,30,25),SMP(30,30,25),
     &            SMPP(30,30,25),PP(30,30,25),DU(30,30,25),
     &            DV(30,30,25),DW(30,30,25)
      COMMON/BL36/AP(30,30,25),AE(30,30,25),AW(30,30,25),AN(30,30,25),
     &            AS(30,30,25),AF(30,30,25),AB(30,30,25),SP(30,30,25),
     &            SU(30,30,25),RI(30,30,25)
      COMMON/BL37/VIS(30,30,25),COND(30,30,25),RESORM(40),
     &            CPM(30,30,25),NHSZ(3,2),NOD(30,30,25)
      COMMON/BL50/JHWALS,JHWALF,KHWALS,KHWALF,NVENT

C *** CALCULATE COEFFICIENTS
      DO 100 K=2,NK+3
      DO 100 J=3,NJ+3
      DO 100 I=2,NI+3

C *** CENTRAL LENGTH OF THE V CONTROL VOLUME
         DXP1=DXXC(I+1)
         DXI =DXXC(I)
         DXM1=DXXC(I-1)

         DYP1=DYYS(J+1)
         DYJ =DYYS(J)
         DYM1=DYYS(J-1)

         DZP1=DZZC(K+1)
         DZK =DZZC(K)
         DZM1=DZZC(K-1)

C *** SURFACE LENGTH OF THE CONTROL VOLUME
         DXN=DXXC(I)
         DXS=DXXC(I)
         DXF=DXXC(I)
         DXB=DXXC(I)

         DYF=DYYS(J)
         DYB=DYYS(J)
         DYE=DYYS(J)
         DYW=DYYS(J)

         DZE=DZZC(K)
         DZW=DZZC(K)
         DZN=DZZC(K)
         DZS=DZZC(K)

C *** CENTRAL LENGTH OF THE STAGGERED CONTROL VOLUME FOR V
         DXEE=DXXS(I+2)
         DXE =DXXS(I+1)
         DXW =DXXS(I)
         DXWW=DXXS(I-1)

         DYNN=DYYC(J+1)
         DYN =DYYC(J)
         DYS =DYYC(J-1)
         DYSS=DYYC(J-2)

         DZFF=DZZC(K+2)
         DZF =DZZC(K+1)
         DZB =DZZC(K)
```

```
         DZBB=DZZC(K-1)

C *** DEFINE THE AREA OF THE CONTROL VOLUME
         DXYF=DXF*DYF
         DXYB=DXB*DYB
         DYZE=DYE*DZE
         DYZW=DYW*DZW
         DZXN=DZN*DXN
         DZXS=DZS*DXS

         VOL  =DXI*DYJ*DZK
         VOLDT=VOL/DTIME

         ZXOYN=DZXN/DYN
         ZXOYS=DZXS/DYS
         XYOZF=DXYF/DZF
         XYOZB=DXYB/DZB
         YZOXE=DYZE/DXE
         YZOXW=DYZW/DXW

C *** USE SINGLE AND BI-LINEAR INTERPOLATION TO EVALUATE
C     PHYSICAL PROPERTIES AND FLUX ON THE SURFACES.
         GEN=SILIN(R(I+1,J  ,K),R(I,J  ,K),DXP1,DXI)*U(I+1,J  ,K)
         GES=SILIN(R(I+1,J-1,K),R(I,J-1,K),DXP1,DXI)*U(I+1,J-1,K)
         GWN=SILIN(R(I-1,J  ,K),R(I,J  ,K),DXM1,DXI)*U(I  ,J  ,K)
         GWS=SILIN(R(I-1,J-1,K),R(I,J-1,K),DXM1,DXI)*U(I  ,J-1,K)

         GN =SILIN(R(I,J+1,K),R(I,J  ,K),DYNN,DYN)*V(1,J+1,K)
         GP =SILIN(R(I,J-1,K),R(I,J  ,K),DYS ,DYN)*V(I,J  ,K)
         GS =SILIN(R(I,J-2,K),R(I,J-1,K),DYSS,DYS)*V(I,J-1,K)

         GFN=SILIN(R(I,J  ,K+1),R(I,J  ,K),DZP1,DZK)*W(I,J  ,K+1)
         GFS=SILIN(R(I,J-1,K+1),R(I,J-1,K),DZP1,DZK)*W(I,J-1,K+1)
         GBN=SILIN(R(I,J  ,K-1),R(I,J  ,K),DZM1,DZK)*W(I,J  ,K  )
         GBS=SILIN(R(I,J-1,K-1),R(I,J-1,K),DZM1,DZK)*W(I,J-1,K  )

C *** MASS FLOW RATE
         CN=0.5*(GN+GP)*DZXN
         CS=0.5*(GP+GS)*DZXS
         CE=SILIN(GEN,GES,DYN,DYS)*DYZE
         CW=SILIN(GWN,GWS,DYN,DYS)*DYZW
         CF=SILIN(GFN,GFS,DYN,DYS)*DXYF
         CB=SILIN(GBN,GBS,DYN,DYS)*DXYB

C *** VISCOSITY
         VISN= VIS(I,J  ,K)
         VISS= VIS(I,J-1,K)
         VISE=(VIS(I+1,J,K)+VIS(I,J,K)+VIS(I+1,J-1,K)+VIS(I,J-1,K))/4.0
         VISW=(VIS(I-1,J,K)+VIS(I,J,K)+VIS(I-1,J-1,K)+VIS(I,J-1,K))/4.0
         VISF=(VIS(I,J,K+1)+VIS(I,J,K)+VIS(I,J-1,K+1)+VIS(I,J-1,K))/4.0
         VISB=(VIS(I,J,K-1)+VIS(I,J,K)+VIS(I,J-1,K-1)+VIS(I,J-1,K))/4.0

         VISN1=ZXOYN*VISN
         VISS1=ZXOYS*VISS
         VISE1=YZOXE*VISE
         VISW1=YZOXW*VISW
         VISF1=XYOZF*VISF
         VISB1=XYOZB*VISB

C *** QUICK SCHEME
         CEP=(ABS(CE)+CE)*DXP1*DXI/(DXE*(DXE+DXW )*8.)
```

```
      CEM=(ABS(CE)-CE)*DXP1*DXI/(DXE*(DXE+DXEE)*8.)
      CWP=(ABS(CW)+CW)*DXM1*DXI/(DXW*(DXW+DXWW)*8.)
      CWM=(ABS(CW)-CW)*DXM1*DXI/(DXW*(DXW+DXE )*8.)

      CNP=(ABS(CN)+CN)*DYN/(DYJ *16.)
      CNM=(ABS(CN)-CN)*DYN/(DYP1*16.)
      CSP=(ABS(CS)+CS)*DYS/(DYM1*16.)
      CSM=(ABS(CS)-CS)*DYS/(DYJ *16.)

      CFP=(ABS(CF)+CF)*DZP1*DZK/(DZF*(DZF+DZB )*8.)
      CFM=(ABS(CF)-CF)*DZP1*DZK/(DZF*(DZF+DZFF)*8.)
      CBP=(ABS(CB)+CB)*DZM1*DZK/(DZB*(DZB+DZBB)*8.)
      CBM=(ABS(CB)-CB)*DZM1*DZK/(DZB*(DZB+DZF )*8.)

      AE(I,J,K)=(-.5*CE*DXI+CWM*DXW)/DXE+CEP+CEM*(1.+DXE/DXEE)+VISE1
      AW(I,J,K)=( .5*CW*DXI+CEP*DXE)/DXW+CWM+CWP*(1.+DXW/DXWW)+VISW1
      AN(I,J,K)= -.5*CN    +CSM*DYS /DYN+CNP+CNM*(1.+DYN/DYNN)+VISN1
      AS(I,J,K)=  .5*CS    +CNP*DYN /DYS+CSM+CSP*(1.+DYS/DYSS)+VISS1
      AF(I,J,K)=(-.5*CF*DZK+CBM*DZB)/DZF+CFP+CFM*(1.+DZF/DZFF)+VISF1
      AB(I,J,K)=( .5*CB*DZK+CFP*DZF)/DZB+CBM+CBP*(1.+DZB/DZBB)+VISB1

C *** BOUNDARY CONSIDERATION
      IF (I.LT.NI+3) THEN
         AEE=-CEM*DXE/DXEE
         AEER=AEE*VPD(I+2,J,K)
      ELSE
         AEE=0.
         AEER=0.
      ENDIF

      IF (I.GT.2) THEN
         AWW=-CWP*DXW/DXWW
         AWWR=AWW*VPD(I-2,J,K)
      ELSE
         AWW=0.
         AWWR=0.
      ENDIF

      IF (J.LT.NJ+3) THEN
         ANN=-CNM*DYN/DYNN
         ANNR=ANN*VPD(I,J+2,K)
      ELSE
         ANN=0.
         ANNR=0.
      ENDIF

      IF (J.GT.3) THEN
         ASS=-CSP*DYS/DYSS
         ASSR=ASS*VPD(I,J-2,K)
      ELSE
         ASS=0.
         ASSR=0.
      ENDIF

      IF (K.LT.NK+3) THEN
         AFF=-CFM*DZF/DZFF
         AFFR=AFF*VPD(I,J,K+2)
      ELSE
         AFF=0.
         AFFR=0.
      ENDIF
```

```
      IF (K.GT.2) THEN
         ABB=-CBP*DZB/DZBB
         ABBR=ABB*VPD(I,J,K-2)
      ELSE
         ABB=0.
         ABBR=0.
      ENDIF

C *** MODIFICATION FOR DECK BOUNDARIES
      IF (NOD(I-1,J,K).NE.0) THEN
         AWW=0.0
         AWWR=0.0
      ENDIF

      IF (NOD(I+1,J,K).NE.0) THEN
         AEE=0.0
         AEER=0.0
      ENDIF

      IF (NOD(I,J-2,K).NE.0) THEN
         ASS=0.0
         ASSR=0.0
      ENDIF

      IF (NOD(I,J+1,K).NE.0) THEN
         ANN=0.0
         ANNR=0.0
      ENDIF

      IF (NOD(I,J,K-1).NE.0) THEN
         ABB=0.0
         ABBR=0.0
      ENDIF

      IF (NOD(I,J,K+1).EQ.0) THEN
         AFF=0.0
         AFFR=0.0
      ENDIF

C *** SU FROM NORMAL STRESS
      RN=(SIG22(I,J  ,K)-(V(I,J+1,K)-V(I,J  ,K))*VISN/DYN)*DZXN
      RS=(SIG22(I,J-1,K)-(V(I,J  ,K)-V(I,J-1,K))*VISS/DYS)*DZXS
      RE=(SIG12(I+1,J,K)-(V(I+1,J,K)-V(I  ,J,K))*VISE/DXE)*DYZE
      RW=(SIG12(I  ,J,K)-(V(I  ,J,K)-V(I-1,J,K))*VISW/DXW)*DYZW
      RF=(SIG23(I,J,K+1)-(V(I,J,K+1)-V(I,J,K  ))*VISF/DZF)*DXYF
      RB=(SIG23(I,J,K  )-(V(I,J,K  )-V(I,J,K-1))*VISB/DZB)*DXYB

C *** SU FROM CURVED STRESSES AND ACCELERATIONS
      AVG12= 0.5*(SIG12(I+1,J,K  )+SIG12(I,J  ,K))
      AVG23= 0.5*(SIG23(I  ,J,K+1)+SIG23(I,J  ,K))
      AVG11=SILIN(SIG11(I  ,J,K  ),SIG11(I,J-1,K),DYN,DYS)
      AVG33=SILIN(SIG33(I  ,J,K  ),SIG33(I,J-1,K),DYN,DYS)

      AU2=V(I,J,K)
      AU1=BILIN(U(I+1,J  ,K  ),U(I,J  ,K),DXI,DXI,
     &          U(I+1,J-1,K  ),U(I,J-1,K),DXI,DXI,DYN,DYS)
      AU3=BILIN(W(I  ,J  ,K+1),W(I,J  ,K),DZK,DZK,
     &          W(I  ,J-1,K+1),W(I,J-1,K),DZK,DZK,DYN,DYS)

      AR=SILIN(R(I,J,K),R(I,J-1,K),DYN,DYS)
```

```
         ARU12=AR*AU1*AU2
         ARU23=AR*AU2*AU3
         ARU11=AR*AU1*AU1
         ARU33=AR*AU3*AU3

         RRX=(AVG12-ARU12)*DZK*(DYE-DYW)
         RRZ=(AVG23-ARU23)*DXI*(DYF-DYB)
         RRY=(AVG11-ARU11)*DZK*(DXN-DXS)+(AVG33-ARU33)*DXI*(DZN-DZS)

         AP(I,J,K)=AE(I,J,K)+AW(I,J,K)+AN(I,J,K)+AS(I,J,K)+
     &            AF(I,J,K)+AB(I,J,K)+AEE+AWW+ANN+ASS+AFF+ABB
         SP(I,J,K)=-(ROD(I,J,K)*DYS+ROD(I,J-1,K)*DYN)*VOLDT/(DYS+DYN)
         SU(I,J,K)=-SP(I,J,K)*VOD(I,J,K)+DZK*DXI*(P(I,J-1,K)-P(I,J,K))+
     &            AEER+AWWR+ANNR+ASSR+AFFR+ABBR+RE-RW+RN-RS+RF-RB+RRX+
     &            RRZ-RRY
  100 CONTINUE

C *** TAKE CARE OF B.C. THRU AN,AS,AE,AW,AF,AB,SP AND SU

C *** Y DIRECTION
      DO 500 K=2,NK+3
      DO 500 I=2,NI+3
         AS(I,3   ,K)=0.
         AN(I,NJ+3,K)=0.
  500 CONTINUE

C *** X DIRECTION
      DO 502 K=2,NK+3
      DO 502 J=3,NJ+3
      IF(NVENT.GT.0) THEN
        IF((J.LE.JHWALS-1 .OR. J.GE.JHWALF+1).OR.
     &     (K.LE.KHWALS-1 .OR. K.GE.KHWALF+1)) THEN
          SP(2   ,J,K)=SP(2   ,J,K)-AW(2   ,J,K)
          AW(2   ,J,K)=0.0
        END IF
      ELSE
          SP(2   ,J,K)=SP(2   ,J,K)-AW(2   ,J,K)
          AW(2   ,J,K)=0.0
      END IF
         SP(NI+3,J,K)=SP(NI+3,J,K)-AE(NI+3,J,K)
         AE(NI+3,J,K)=0.0
  502 CONTINUE
       IF(NVENT.LT.0) GOTO 598
       DO 503 K=KHWALS,KHWALF
       DO 503 J=JHWALS,JHWALF
         SP(2   ,J,K)=SP(2   ,J,K)+AW(2   ,J,K)
         AW(2   ,J,K)=0.0
  503  CONTINUE
  598  CONTINUE
C *** Z DIRECTION
      DO 600 I=2,NI+3
      DO 600 J=3,NJ+3
         SP(I,J,2   )=SP(I,J,2   )-AB(I,J,2   )
         SP(I,J,NK+3)=SP(I,J,NK+3)-AF(I,J,NK+3)
         AF(I,J,NK+3)=0.
         AB(I,J,2   )=0.
  600 CONTINUE

C *** MODIFICATION FOR DECK BOUNDARIES
      IF (NCHIP.EQ.0) GOTO 201
```

```
      DO 101 N=1,NCHIP
         IB  =ICHPB(N)
         IE  =IB+NCHPI(N)-1
         JB  =JCHPB(N)
         JE  =JB+NCHPJ(N)-1
         KB  =KCHPB(N)
         KE  =KB+NCHPK(N)-1
         DO 102 J=JB,JE
         DO 102 K=KB,KE-1
            SP(IB-1,J,K)=SP(IB-1,J,K)-AE(IB-1,J,K)
            AE(IB-1,J,K)=0.0
            SP(IE  ,J,K)=SP(IE  ,J,K)-AW(IE  ,J,K)
            AW(IE  ,J,K)=0.0
 102     CONTINUE

         DO 103 I=IB,IE-1
         DO 103 K=KB,KE-1
            AN(I,JB-1,K)=0.0
            AS(I,JE+1,K)=0.0
 103     CONTINUE

         DO 106 I=IB,IE-1
         DO 106 J=JB,JE
            SP(I,J,KB-1)=SP(I,J,KB-1)-AF(I,J,KB-1)
            AF(I,J,KB-1)=0.0
            SP(I,J,KE  )=SP(I,J,KE  )-AB(I,J,KE  )
            AB(I,J,KE  )=0.0
 106     CONTINUE

C *** MODIFICATION FOR THE CELLS INSIDE OF THE DECKS
         DO 104 I=IB,IE-1
         DO 104 J=JB,JE
         DO 104 K=KB,KE-1
            SP(I,J,K)=-1.0E2
            AW(I,J,K)=0.
            AE(I,J,K)=0.
            AS(I,J,K)=0.
            AN(I,J,K)=0.
            AB(I,J,K)=0.
            AF(I,J,K)=0.
            SU(I,J,K)=0.
 104     CONTINUE

 101  CONTINUE

C *** ASSEMBLE COEFFICIENTS AND SOLVE DIFFERENCE EQUATIONS
 201  DO 300 K=2,NK+3
      DO 300 J=3,NJ+3
      DO 300 I=2,NI+3
         DXI=DXXC(I)
         DZK=DZZC(K)
         DZX=DZK*DXI
         AP(I,J,K)=AP(I,J,K)-SP(I,J,K)
         DV(I,J,K)=DZX/AP(I,J,K)
 300  CONTINUE

C *** SOLVE FOR V
      CALL TRID (3,4,3,NI+2,NJ+2,NK+2,V)


C *** RESET THE VELOCITY INSIDE OF THE DECKS
```

```
      IF (NCHIP.EQ.0) GOTO 111
      DO 110 N=1,NCHIP
         IB=ICHPB(N)
         IE=IB+NCHPI(N)-1
         JB=JCHPB(N)
         JE=JB+NCHPJ(N)-1
         KB=KCHPB(N)
         KE=KB+NCHPK(N)-1
         DO 108 I=IB,IE-1
         DO 108 J=JB,JE
         DO 108 K=KB,KE-1
            V(I,J,K)=0.0
  108    CONTINUE
  110 CONTINUE

  111 RETURN
      END

************************************************************************
************************************************************************
      SUBROUTINE CALVIS
************************************************************************
*     THIS SUBROUTINE CALCULATES THE TURBULENT VISCOSITY AND UPDATES   *
*     THE VISCOSITY MATRIX                                             *
************************************************************************

      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      COMMON/R4/XC(40),YC(40),ZC(40),XS(40),YS(40),ZS(40),DXXC(40),
     &          DYYC(40),DZZC(40),DXXS(40),DYYS(40),DZZS(40)
      COMMON/BL7/NI,NJ,NK,KRUN,NBLOR,NWRP
      COMMON/BL14/HCOEF,CNT,ABTURB,BTURB,VISL,VISMAX
      COMMON/BL16/U0,UGRT,BUOY,CP0,PRT,COND0,VIS0,RHO0,
     &            TA,DTEMP,TWRITE,TTAPE,TMAX,GC,RAIR,NT
      COMMON/BL32/T(30,30,25),R(30,30,25),P(30,30,25),C(30,30,25),
     &            U(30,30,25),V(30,30,25),W(30,30,25)
      COMMON/BL34/HEIGHT(30,30,25),REQ(30,30,25),SMP(30,30,25),
     &            SMPP(30,30,25),PP(30,30,25),DU(30,30,25),
     &            DV(30,30,25),DW(30,30,25)
      COMMON/BL36/AP(30,30,25),AE(30,30,25),AW(30,30,25),AN(30,30,25),
     &            AS(30,30,25),AF(30,30,25),AB(30,30,25),SP(30,30,25),
     &            SU(30,30,25),RI(30,30,25)
      COMMON/BL37/VIS(30,30,25),COND(30,30,25),RESORM(40),
     &            CPM(30,30,25),NHSZ(3,2),NOD(30,30,25)
      COMMON/BL50/JHWALS,JHWALF,KHWALS,KHWALF,NVENT

C *** CALCULATE LOCAL SHEAR AND VISCOSITY VIS(I,J,K)

C *** SPECIFY LOCAL TURBULENT LENGTH SCALES SMPP(I,J,K)
      DO 611 K=3,NK+2
      DO 611 J=3,NJ+2
      DO 611 I=3,NI+2

C *** CENTRAL LENGTH OF THE SCALAR CONTROL VOLUME
         DXP1=DXXC(I+1)
         DXI =DXXC(I)
         DXM1=DXXC(I-1)

         DYP1=DYYC(J+1)
         DYJ =DYYC(J)
         DYM1=DYYC(J-1)
```

```
      DZP1=DZZC(K+1)
      DZK =DZZC(K)
      DZM1=DZZC(K-1)

C *** CENTRAL LENGTH OF THE STAGGERED CONTROL VOLUME FOR T
      DXE =DXXS(I+1)
      DXW =DXXS(I)

      DYN =DYYS(J+1)
      DYS =DYYS(J)

      DZF =DZZS(K+1)
      DZB =DZZS(K)

C *** CACULATE DV/DX,D2V/DX2,DU/DX,D2U/DX2,DW/DX AND D2W/DX2
      DUDX  =    (U(I+1,J,K)-U(I  ,J,K))/DXI
      DUDXW =0.5*(U(I+1,J,K)-U(I-1,J,K))/DXW
      DUDXE =0.5*(U(I+2,J,K)-U(I  ,J,K))/DXE
      D2UDX2=(DUDXE-DUDXW)/DXI

      DVDXW =0.5*(V(I,J+1,K)+V(I,J,K)-V(I-1,J+1,K)-V(I-1,J,K))/DXW
      DVDXE =0.5*(V(I+1,J+1,K)+V(I+1,J,K)-V(I,J+1,K)-V(I,J,K))/DXE
      DVDX  =0.5*(DVDXE+DVDXW)
      D2VDX2=    (DVDXE-DVDXW)/DXI

      DWDXW =0.5*(W(I,J,K+1)+W(I,J,K)-W(I-1,J,K+1)-W(I-1,J,K))/DXW
      DWDXE =0.5*(W(I+1,J,K+1)+W(I+1,J,K)-W(I,J,K+1)-W(I,J,K))/DXE
      DWDX  =0.5*(DWDXE+DWDXW)
      D2WDX2=    (DWDXE-DWDXW)/DXI

C *** CALCULATE DU/DY,D2U/DY2,DV/DY,D2V/DY2,DW/DY AND D2W/DY2
      DVDY  =    (V(I,J+1,K)-V(I,J  ,K))/DYJ
      DVDYS =0.5*(V(I,J+1,K)-V(I,J-1,K))/DYS
      DVDYN =0.5*(V(I,J+2,K)-V(I,J  ,K))/DYN
      D2VDY2=(DVDYN-DVDYS)/DYJ

      DUDYS =0.5*(U(I+1,J,K)+U(I,J,K)-U(I+1,J-1,K)-U(I,J-1,K))/DYS
      DUDYN =0.5*(U(I+1,J+1,K)+U(I,J+1,K)-U(I+1,J,K)-U(I,J,K))/DYN
      DUDY  =0.5*(DUDYN+DUDYS)
      D2UDY2=    (DUDYN-DUDYS)/DYJ

      DWDYS =0.5*(W(I,J,K+1)+W(I,J,K)-W(I,J-1,K+1)-W(I,J-1,K))/DYS
      DWDYN =0.5*(W(I,J+1,K+1)+W(I,J+1,K)-W(I,J,K+1)-W(I,J,K))/DYN
      DWDY  =0.5*(DWDYN+DWDYS)
      D2WDY2=    (DWDYN-DWDYS)/DYJ

C *** CALCULATE DU/DZ,D2U/DZ2,DV/DZ,D2V/DZ2,DW/DZ AND D2W/DZ2
      DWDZ  =    (W(I,J,K+1)-W(I,J,K  ))/DZK
      DWDZF =0.5*(W(I,J,K+2)-W(I,J,K  ))/DZF
      DWDZB =0.5*(W(I,J,K+1)-W(I,J,K-1))/DZB
      D2WDZ2=(DWDZF-DWDZB)/DZK

      DVDZB =0.5*(V(I,J+1,K)+V(I,J,K)-V(I,J+1,K-1)-V(I,J,K-1))/DZB
      DVDZF =0.5*(V(I,J+1,K+1)+V(I,J,K+1)-V(I,J+1,K)-V(I,J,K))/DZF
      DVDZ  =0.5*(DVDZF+DVDZB)
      D2VDZ2=    (DVDZF-DVDZB)/DZK

      DUDZB =0.5*(U(I+1,J,K)+U(I,J,K)-U(I+1,J,K-1)-U(I,J,K-1))/DZB
      DUDZF =0.5*(U(I+1,J,K+1)+U(I,J,K+1)-U(I+1,J,K)-U(I,J,K))/DZF
      DUDZ  =0.5*(DUDZF+DUDZB)
      D2UDZ2=    (DUDZF-DUDZB)/DZK
```

```
C *** CALCULATE THE DENSITY GRADIENT WITH RESPECT TO THE VERTICAL
         DRDGA=(R(I,J,K+1)-REQ(I,J,K+1)-R(I,J,K-1)+REQ(I,J,K-1))/
     &           (DZF+DZB)

C *** CALCULATE STRAIN
         STRAIN=DUDY**2+DVDX**2+DWDX**2+DVDZ**2+DWDY**2+DUDZ**2
         DDO2   =SQRT(STRAIN+DUDX**2+DVDY**2+DWDZ**2)

         IF(DDO2.EQ.0..OR.STRAIN.EQ.0.) THEN
            VIS(I,J,K)=VISL
         ELSE

C *** CALCULATE TURBULENT LENGTH SCALE SMPP(I,J)
            SMP123=SQRT(((U(I+1,J,K)+U(I,J,K))/2.)**2+
     &                  ((V(I,J+1,K)+V(I,J,K))/2.)**2+
     &                  ((W(I,J,K+1)+W(I,J,K))/2.)**2)/DDO2
            SMPP12=DDO2/SQRT(D2UDX2**2+D2UDY2**2+D2UDZ2**2+D2VDX2**2+
     &                D2VDY2**2+D2VDZ2**2+D2WDZ2**2+D2WDX2**2+D2WDY2**2)
            SMPP(I,J,K)=CNT*(SMP123+SMPP12)/2.

C *** CALCULATE RICHARDSON NUMBER
            RI(I,J,K)=-BUOY*DRDGA/(R(I,J,K)*STRAIN)
            ABRIPR=ABTURB+RI(I,J,K)/PRT

            IF(ABRIPR.LT.0.) THEN
               VIS(I,J,K)=VISL
            ELSEIF(ABRIPR.EQ.0.) THEN
               VIS(I,J,K)=VISMAX
            ELSE
               VIS(I,J,K)=VISL+R(I,J,K)*SMPP(I,J,K)**2*
     &                    SQRT(STRAIN)/(BTURB*ABRIPR)
               IF(VIS(I,J,K).GT.VISMAX) VIS(I,J,K)=VISMAX
            ENDIF
         ENDIF
  611 CONTINUE

C *** SPECIFY THE VISCOCITY ON THE BOUNDARY POINT
      DO 110 I=1,NI+4
      DO 110 J=1,NJ+4
         VIS(I,J,NK+3)=VIS(I,J,NK+2)
         VIS(I,J,2   )=VIS(I,J,3   )
  110 CONTINUE

      DO 120 J=1,NJ+4
      DO 120 K=1,NK+4
         VIS(NI+3,J,K)=VIS(NI+2,J,K)
         VIS(2   ,J,K)=VIS(3   ,J,K)
  120 CONTINUE

      DO 130 K=1,NK+4
      DO 130 I=1,NI+4
         VIS(I,NJ+3,K)=VIS(I,NJ+2,K)
         VIS(I,2   ,K)=VIS(I,3   ,K)
  130 CONTINUE

C *** CALCULATE TURBULENT CONDUCTIVITY
      DO 140 I=1,NI+4
      DO 140 J=1,NJ+4
      DO 140 K=1,NK+4
         IF (NOD(I,J,K).NE.1) COND(I,J,K)=VIS(I,J,K)/PRT
```

```
  140 CONTINUE
      RETURN
      END

************************************************************************
************************************************************************
      SUBROUTINE CALW
************************************************************************
*CALCULATES THE W COMPONENT OF THE VELOCITY

      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      COMMON/R4/XC(40),YC(40),ZC(40),XS(40),YS(40),ZS(40),DXXC(40),
     &         DYYC(40),DZZC(40),DXXS(40),DYYS(40),DZZS(40)
      COMMON/BL1/DX,DY,DZ,DTIME,TCOOL,PI,Q,QR
      COMMON/BL7/NI,NJ,NK,KRUN,NBLOR,NWRP
      COMMON/BL16/U0,UGRT,BUOY,CP0,PRT,COND0,VIS0,RHO0,
     &            TA,DTEMP,TWRITE,TTAPE,TMAX,GC,RAIR,NT
      COMMON/BL20/SIG11(30,30,25),SIG12(30,30,25),SIG22(30,30,25),
     &            SIG13(30,30,25),SIG23(30,30,25),SIG33(30,30,25)
      COMMON/BL22/CPS(20),CONS(20),WFAN(20),NCHIP,ICHPB(20),NCHPI(20),
     &            JCHPB(20),NCHPJ(20),KCHPB(20),NCHPK(20)
      COMMON/BL31/TOD(30,30,25),ROD(30,30,25),POD(30,30,25),
     &            COD(30,30,25),UOD(30,30,25),VOD(30,30,25),
     &            WOD(30,30,25)
      COMMON/BL32/T(30,30,25),R(30,30,25),P(30,30,25),C(30,30,25),
     &            U(30,30,25),V(30,30,25),W(30,30,25)
      COMMON/BL33/TPD(30,30,25),RPD(30,30,25),PPD(30,30,25),
     &            CPD(30,30,25),UPD(30,30,25),VPD(30,30,25),
     &            WPD(30,30,25)
      COMMON/BL34/HEIGHT(30,30,25),REQ(30,30,25),SMP(30,30,25),
     &            SMPP(30,30,25),PP(30,30,25),DU(30,30,25),
     &            DV(30,30,25),DW(30,30,25)
      COMMON/BL36/AP(30,30,25),AE(30,30,25),AW(30,30,25),AN(30,30,25),
     &            AS(30,30,25),AF(30,30,25),AB(30,30,25),SP(30,30,25),
     &            SU(30,30,25),RI(30,30,25)
      COMMON/BL37/VIS(30,30,25),COND(30,30,25),RESORM(40),
     &            CPM(30,30,25),NHSZ(3,2),NOD(30,30,25)
      COMMON/BL50/JHWALS,JHWALF,KHWALS,KHWALF,NVENT

C *** CALCULATE COEFFICIENTS
      DO 100 K=3,NK+3
      DO 100 J=2,NJ+3
      DO 100 I=2,NI+3

C *** CENTRAL LENGTH OF THE W CONTROL VOLUME
         DXP1=DXXC(I+1)
         DXI =DXXC(I)
         DXM1=DXXC(I-1)

         DYP1=DYYC(J+1)
         DYJ =DYYC(J)
         DYM1=DYYC(J-1)

         DZP1=DZZS(K+1)
         DZK =DZZS(K)
         DZM1=DZZS(K-1)

C *** SURFACE LENGTH OF THE CONTROL VOLUME
         DXN=DXXC(I)
         DXS=DXXC(I)
         DXF=DXXC(I)
```

```
      DXB=DXXC(I)

      DYF=DYYC(J)
      DYB=DYYC(J)
      DYE=DYYC(J)
      DYW=DYYC(J)

      DZE=DZZS(K)
      DZW=DZZS(K)
      DZN=DZZS(K)
      DZS=DZZS(K)

C *** CENTRAL LENGTH OF THE STAGGERED CONTROL VOLUME
      DXEE=DXXS(I+2)
      DXE =DXXS(I+1)
      DXW =DXXS(I)
      DXWW=DXXS(I-1)

      DYNN=DYYS(J+2)
      DYN =DYYS(J+1)
      DYS =DYYS(J)
      DYSS=DYYS(J-1)

      DZFF=DZZC(K+1)
      DZF =DZZC(K)
      DZB =DZZC(K-1)
      DZBB=DZZC(K-2)

C *** DEFINE THE AREA OF THE CONTROL VOLUME
      DXYF=DXF*DYF
      DXYB=DXB*DYB
      DYZE=DYE*DZE
      DYZW=DYW*DZW
      DZXN=DZN*DXN
      DZXS=DZS*DXS

      VOL=DXI*DYJ*DZK
      VOLDT=VOL/DTIME

      ZXOYN=DZXN/DYN
      ZXOYS=DZXS/DYS
      XYOZF=DXYF/DZF
      XYOZB=DXYB/DZB
      YZOXE=DYZE/DXE
      YZOXW=DYZW/DXW

C *** USE SINGLE AND BI-LINEAR INTERPOLATION TO EVALUATE
C     PHYSICAL PROPERTIES AND FLUX ON THE SURFACES.
      GNF=SILIN(R(I,J+1,K  ),R(I,J,K  ),DYP1,DYJ)*V(I,J+1,K  )
      GNB=SILIN(R(I,J+1,K-1),R(I,J,K-1),DYP1,DYJ)*V(I,J+1,K-1)
      GSF=SILIN(R(I,J-1,K  ),R(I,J,K  ),DYM1,DYJ)*V(I,J  ,K  )
      GSB=SILIN(R(I,J-1,K-1),R(I,J,K-1),DYM1,DYJ)*V(I,J  ,K-1)

      GF =SILIN(R(I,J,K+1),R(I,J,K  ),DZFF,DZF)*W(I,J,K+1)
      GP =SILIN(R(I,J,K-1),R(I,J,K  ),DZB ,DZF)*W(I,J,K  )
      GB =SILIN(R(I,J,K-2),R(I,J,K-1),DZBB,DZB)*W(I,J,K-1)

      GEF=SILIN(R(I+1,J,K  ),R(I,J,K  ),DXP1,DXI)*U(I+1,J,K  )
      GEB=SILIN(R(I+1,J,K-1),R(I,J,K-1),DXP1,DXI)*U(I+1,J,K-1)
      GWF=SILIN(R(I-1,J,K  ),R(I,J,K  ),DXM1,DXI)*U(I  ,J,K  )
      GWB=SILIN(R(I-1,J,K-1),R(I,J,K-1),DXM1,DXI)*U(I  ,J,K-1)
```

```
C *** MASS FLOW RATE
          CN=SILIN(GNF,GNB,DZF,DZB)*DZXN
          CS=SILIN(GSF,GSB,DZF,DZB)*DZXS
          CE=SILIN(GEF,GEB,DZF,DZB)*DYZE
          CW=SILIN(GWF,GWB,DZF,DZB)*DYZW
          CF=0.5*(GF+GP)*DXYF
          CB=0.5*(GP+GB)*DXYB

C *** VISCOSITY
          VISF=VIS(I,J,K  )
          VISB=VIS(I,J,K-1)
          VISN=(VIS(I,J+1,K)+VIS(I,J,K)+VIS(I,J+1,K-1)+VIS(I,J,K-1))/4.
          VISS=(VIS(I,J-1,K)+VIS(I,J,K)+VIS(I,J-1,K-1)+VIS(I,J,K-1))/4.
          VISE=(VIS(I+1,J,K)+VIS(I,J,K)+VIS(I+1,J,K-1)+VIS(I,J,K-1))/4.
          VISW=(VIS(I-1,J,K)+VIS(I,J,K)+VIS(I-1,J,K-1)+VIS(I,J,K-1))/4.

          VISN1=ZXOYN*VISN
          VISS1=ZXOYS*VISS
          VISE1=YZOXE*VISE
          VISW1=YZOXW*VISW
          VISF1=XYOZF*VISF
          VISB1=XYOZB*VISB

C *** QUICK SCHEME
          CEP=(ABS(CE)+CE)*DXP1*DXI/(8.*DXE*(DXE+DXW ))
          CEM=(ABS(CE)-CE)*DXP1*DXI/(8.*DXE*(DXE+DXEE))
          CWP=(ABS(CW)+CW)*DXM1*DXI/(8.*DXW*(DXW+DXWW))
          CWM=(ABS(CW)-CW)*DXM1*DXI/(8.*DXW*(DXW+DXE ))

          CNP=(ABS(CN)+CN)*DYP1*DYJ/(8.*DYN*(DYN+DYS ))
          CNM=(ABS(CN)-CN)*DYP1*DYJ/(8.*DYN*(DYN+DYNN))
          CSP=(ABS(CS)+CS)*DYM1*DYJ/(8.*DYS*(DYS+DYSS))
          CSM=(ABS(CS)-CS)*DYM1*DYJ/(8.*DYS*(DYS+DYN ))

          CFP=(ABS(CF)+CF)*DZF/(DZK *16.)
          CFM=(ABS(CF)-CF)*DZF/(DZP1*16.)
          CBP=(ABS(CB)+CB)*DZB/(DZM1*16.)
          CBM=(ABS(CB)-CB)*DZB/(DZK *16.)

          AE(I,J,K)=(-.5*CE*DXI+CWM*DXW)/DXE+CEP+CEM*(1.+DXE/DXEE)+VISE1
          AW(I,J,K)=( .5*CW*DXI+CEP*DXE)/DXW+CWM+CWP*(1.+DXW/DXWW)+VISW1
          AN(I,J,K)=(-.5*CN*DYJ+CSM*DYS)/DYN+CNP+CNM*(1.+DYN/DYNN)+VISN1
          AS(I,J,K)=( .5*CS*DYJ+CNP*DYN)/DYS+CSM+CSP*(1.+DYS/DYSS)+VISS1
          AF(I,J,K)= -.5*CF     +CBM*DZB /DZF+CFP+CFM*(1.+DZF/DZFF)+VISF1
          AB(I,J,K)=  .5*CB     +CFP*DZF /DZB+CBM+CBP*(1.+DZB/DZBB)+VISB1

C *** BOUNDARY CONSIDERATION
          IF (I.LT.NI+3) THEN
             AEE=-CEM*DXE/DXEE
             AEER=AEE*WPD(I+2,J,K)
          ELSE
             AEE=0.
             AEER=0.
          ENDIF

          IF (I.GT.2) THEN
             AWW=-CWP*DXW/DXWW
             AWWR=AWW*WPD(I-2,J,K)
          ELSE
             AWW=0.
```

```
            AWWR=0.
         ENDIF

         IF (J.LT.NJ+3) THEN
            ANN=-CNM*DYN/DYNN
            ANNR=ANN*WPD(I,J+2,K)
         ELSE
            ANN=0.
            ANNR=0.
         ENDIF

         IF (J.GT.2) THEN
            ASS=-CSP*DYS/DYSS
            ASSR=ASS*WPD(I,J-2,K)
         ELSE
            ASS=0.
            ASSR=0.
         ENDIF

         IF (K.LT.NK+3) THEN
            AFF=-CFM*DZF/DZFF
            AFFR=AFF*WPD(I,J,K+2)
         ELSE
            AFF=0.
            AFFR=0.
         ENDIF

         IF (K.GT.3) THEN
            ABB=-CBP*DZB/DZBB
            ABBR=ABB*WPD(I,J,K-2)
         ELSE
            ABB=0.
            ABBR=0.
         ENDIF

C *** MODIFICATION FOR DECK BOUNDARIES
         IF (NOD(I-1,J,K).NE.0) THEN
            AWW=0.0
            AWWR=0.0
         ENDIF

         IF (NOD(I+1,J,K).NE.0) THEN
            AEE=0.0
            AEER=0.0
         ENDIF

         IF (NOD(I,J-1,K).NE.0) THEN
            ASS=0.0
            ASSR=0.0
         ENDIF

         IF (NOD(I,J+1,K).NE.0) THEN
            ANN=0.0
            ANNR=0.0
         ENDIF

         IF (NOD(I,J,K-2).NE.0) THEN
            ABB=0.0
            ABBR=0.0
         ENDIF
```

```
          IF (NOD(I,J,K+1).NE.0) THEN
             AFF=0.0
             AFFR=0.0
          ENDIF

C *** SU FROM NORMAL STRESS
          RF=(SIG33(I,J,K  )-(W(I,J,K+1)-W(I,J,K  ))*VISF/DZF)*DXYF
          RB=(SIG33(I,J,K-1)-(W(I,J,K  )-W(I,J,K-1))*VISB/DZB)*DXYB
          RN=(SIG23(I,J+1,K)-(W(I,J+1,K)-W(I,J  ,K))*VISN/DYN)*DZXN
          RS=(SIG23(I,J  ,K)-(W(I,J  ,K)-W(I,J-1,K))*VISS/DYS)*DZXS
          RE=(SIG13(I+1,J,K)-(W(I+1,J,K)-W(I  ,J,K))*VISE/DXE)*DYZE
          RW=(SIG13(I  ,J,K)-(W(I  ,J,K)-W(I-1,J,K))*VISW/DXW)*DYZW

C *** SU FROM CURVED STRESSES AND ACCELERATIONS
          AVG23= 0.5*(SIG23(I  ,J+1,K)+SIG23(I,J,K))
          AVG13= 0.5*(SIG13(I+1,J  ,K)+SIG13(I,J,K))
          AVG22=SILIN(SIG22(I,J,K),SIG22(I,J,K-1),DZF,DZB)
          AVG11=SILIN(SIG11(I,J,K),SIG11(I,J,K-1),DZF,DZB)

          AU3=W(I,J,K)
          AU2=BILIN(V(I,J+1,K  ),V(I,J,K  ),DYJ,DYJ,
     &              V(I,J+1,K-1),V(I,J,K-1),DYJ,DYJ,DZF,DZB)
          AU1=BILIN(U(I+1,J,K  ),U(I,J,K  ),DXI,DXI,
     &              U(I+1,J,K-1),U(I,J,K-1),DXI,DXI,DZF,DZB)

          AR=SILIN(R(I,J,K),R(I,J,K-1),DZF,DZB)

          ARU23=AR*AU2*AU3
          ARU13=AR*AU1*AU3
          ARU22=AR*AU2*AU2
          ARU11=AR*AU1*AU1

          RRY=(AVG23-ARU23)*DXI*(DZN-DZS)
          RRX=(AVG13-ARU13)*DYJ*(DZE-DZW)
          RRZ=(AVG22-ARU22)*DXI*(DYF-DYB)+(AVG11-ARU11)*DYJ*(DXF-DXB)

          AP(I,J,K)=AE(I,J,K)+AW(I,J,K)+AN(I,J,K)+AS(I,J,K)
     &              +AF(I,J,K)+AB(I,J,K)+AEE+AWW+ANN+ASS+AFF+ABB
          SP(I,J,K)=-(ROD(I,J,K)*DZB+ROD(I,J,K-1)*DZF)*VOLDT/(DZB+DZF)
          SU(I,J,K)=-SP(I,J,K)*WOD(I,J,K)+DXI*DYJ*(P(I,J,K-1)-P(I,J,K))+
     &              AEER+AWWR+ANNR+ASSR+AFFR+ABBR+RE-RW+RN-RS+RF-RB+RRY+
     &              RRX-RRZ-BUOY*((R(I,J,K)-REQ(I,J,K))*DZB+(R(I,J,K-1)-
     &              REQ(I,J,K-1))*DZF)*VOL/(DZB+DZF)
 100  CONTINUE

C *** TAKE CARE OF B.C. THRU AN,AS,AE,AW,AP AND SU

C *** Y DIRECTION
      DO 500 K=3,NK+3
      DO 500 I=2,NI+3
          SP(I,2   ,K)=SP(I,2   ,K)-AS(I,2   ,K)
          SP(I,NJ+3,K)=SP(I,NJ+3,K)-AN(I,NJ+3,K)
          AS(I,2   ,K)=0.
          AN(I,NJ+3,K)=0.
 500  CONTINUE

C *** X DIRECTION
      DO 502 K=2,NK+3
      DO 502 J=3,NJ+3
      IF(NVENT.GT.0) THEN
        IF((J.LE.JHWALS-1 .OR. J.GE.JHWALF+1).OR.
```

```
     &    (K.LE.KHWALS-1 .OR. K.GE.KHWALF+1)) THEN
            SP(2    ,J,K)=SP(2    ,J,K)-AW(2    ,J,K)
            AW(2    ,J,K)=0.0
          END IF
        ELSE
           SP(2    ,J,K)=SP(2    ,J,K)-AW(2    ,J,K)
           AW(2    ,J,K)=0.0
        END IF
           SP(NI+3,J,K)=SP(NI+3,J,K)-AE(NI+3,J,K)
           AE(NI+3,J,K)=0.0
  502 CONTINUE
         IF(NVENT.LT.0) GOTO 598
         DO 503 K=KHWALS,KHWALF
         DO 503 J=JHWALS,JHWALF
           SP(2    ,J,K)=SP(2    ,J,K)+AW(2    ,J,K)
           AW(2    ,J,K)=0.0
  503  CONTINUE
  598  CONTINUE
C *** Z DIRECTION
        DO 600 I=2,NI+3
        DO 600 J=2,NJ+3
           AF(I,J,NK+3)=0.
           AB(I,J,3   )=0.
  600 CONTINUE

C *** MODIFICATION FOR DECK BOUNDARIES
        IF (NCHIP.EQ.0) GOTO 201
        DO 101 N=1,NCHIP
           IB  =ICHPB(N)
           IE  =IB+NCHPI(N)-1
           JB  =JCHPB(N)
           JE  =JB+NCHPJ(N)-1
           KB  =KCHPB(N)
           KE  =KB+NCHPK(N)-1

           DO 102 J=JB,JE-1
           DO 102 K=KB,KE
              SP(IB-1,J,K)=SP(IB-1,J,K)-AE(IB-1,J,K)
              SP(IE  ,J,K)=SP(IE  ,J,K)-AW(IE  ,J,K)
              SU(IB-1,J,K)=SU(IB-1,J,K)+AE(IB-1,J,K)*WFAN(N)*2.0
              SU(IE  ,J,K)=SU(IE  ,J,K)+AW(IE  ,J,K)*WFAN(N)*2.0
              AE(IB-1,J,K)=0.0
              AW(IE  ,J,K)=0.0
  102      CONTINUE

           DO 103 I=IB,IE-1
           DO 103 K=KB,KE
              SP(I,JB-1,K)=SP(I,JB-1,K)-AN(I,JB-1,K)
              SP(I,JE  ,K)=SP(I,JE  ,K)-AS(I,JE  ,K)
              SU(I,JB-1,K)=SU(I,JB-1,K)+AN(I,JB-1,K)*WFAN(N)*2.0
              SU(I,JE  ,K)=SU(I,JE  ,K)+AS(I,JE  ,K)*WFAN(N)*2.0
              AN(I,JB-1,K)=0.0
              AS(I,JE  ,K)=0.0
  103 CONTINUE

           DO 106 I=IB,IE-1
           DO 106 J=JB,JE-1
              SU(I,J,KB-1)=SU(I,J,KB-1)+AF(I,J,KB-1)*WFAN(N)
              SU(I,J,KE+1)=SU(I,J,KE+1)+AB(I,J,KE+1)*WFAN(N)
              AF(I,J,KB-1)=0.0
              AB(I,J,KE+1)=0.0
```

```
  106     CONTINUE

C *** FOR THE CELLS INSIDE OF THE DECKS
          DO 104 I=IB,IE-1
          DO 104 J=JB,JE-1
          DO 104 K=KB,KE
             SP(I,J,K)=-1.0E2
             AW(I,J,K)=0.
             AE(I,J,K)=0.
             AS(I,J,K)=0.
             AN(I,J,K)=0.
             AB(I,J,K)=0.
             AF(I,J,K)=0.
             SU(I,J,K)=1.0E2*WFAN(N)
  104     CONTINUE
  101 CONTINUE

C *** ASSEMBLE COEFFICIENTS AND SOLVE DIFFERENCE EQUATIONS
  201 DO 301 K=3,NK+3
      DO 301 J=2,NJ+3
      DO 301 I=2,NI+3
         DXI=DXXC(I)
         DYJ=DYYC(J)
         DXY=DXI*DYJ
         AP(I,J,K)=AP(I,J,K)-SP(I,J,K)
         DW(I,J,K)=DXY/AP(I,J,K)
  301 CONTINUE

C *** SOLVE FOR W
      CALL TRID  (3,3,4,NI+2,NJ+2,NK+2,W)

C **** BOUNDARY CONDITIONS DUE TO HOLE IN WEST WALL ****

C *** RESET THE VELOCITY INSIDE OF THE DECKS
      IF (NCHIP.EQ.0) GOTO 111
      DO 110 N=1,NCHIP
         IB=ICHPB(N)
         IE=IB+NCHPI(N)-1
         JB=JCHPB(N)
         JE=JB+NCHPJ(N)-1
         KB=KCHPB(N)
         KE=KB+NCHPK(N)-1
         DO 108 I=IB,IE-1
         DO 108 J=JB,JE-1
         DO 108 K=KB,KE
            W(I,J,K)=WFAN(N)
  108    CONTINUE
  110 CONTINUE

  111 RETURN
      END

***********************************************************************
***********************************************************************
      SUBROUTINE GLOBE
***********************************************************************
*THIS SUBROUTINE CALCULATES THE GLOBAL PRESSURE CORRECTION, WHEREBY THE
*PRESSURE MATRIX IS UPDATED.
*
*VARIABLES USED ARE:
* SUMT   = SUM OF TEMPERATURES
```

```
* SUMPT  = SUM OF PRESSURE OVER TEMPERATURE
* SUMPET = SUM OF EQUILIBRIUM PRESSURE OVER TEMP
* UGRT   = CONSTANT (FROM SUBROUTINE INIT)
* PCORR  = PRESSURE CORRECTION
***********************************************************************

      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      COMMON/R4/XC(40),YC(40),ZC(40),XS(40),YS(40),ZS(40),DXXC(40),
     &          DYYC(40),DZZC(40),DXXS(40),DYYS(40),DZZS(40)
      COMMON/BL1/DX,DY,DZ,DTIME,TCOOL,PI,Q,QR
      COMMON/BL7/NI,NJ,NK,KRUN,NBLOR,NWRP
      COMMON/BL16/U0,UGRT,BUOY,CP0,PRT,COND0,VIS0,RHO0,
     &            TA,DTEMP,TWRITE,TTAPE,TMAX,GC,RAIR,NT
      COMMON/BL31/TOD(30,30,25),ROD(30,30,25),POD(30,30,25),
     &            COD(30,30,25),UOD(30,30,25),VOD(30,30,25),
     &            WOD(30,30,25)
      COMMON/BL32/T(30,30,25),R(30,30,25),P(30,30,25),C(30,30,25),
     &            U(30,30,25),V(30,30,25),W(30,30,25)
      COMMON/BL33/TPD(30,30,25),RPD(30,30,25),PPD(30,30,25),
     &            CPD(30,30,25),UPD(30,30,25),VPD(30,30,25),
     &            WPD(30,30,25)
      COMMON/BL34/HEIGHT(30,30,25),REQ(30,30,25),SMP(30,30,25),
     &            SMPP(30,30,25),PP(30,30,25),DU(30,30,25),
     &            DV(30,30,25),DW(30,30,25)
      COMMON/BL37/VIS(30,30,25),COND(30,30,25),RESORM(40),
     &            CPM(30,30,25),NHSZ(3,2),NOD(30,30,25)
      COMMON/BL50/JHWALS,JHWALF,KHWALS,KHWALF,NVENT
      COMMON/BL52/CONVMT

      SUMT=0.
      SUMPT=0.
      SUMPET=0.
      DO 370 I=3,NI+2
      DO 370 J=3,NJ+2
      DO 370 K=3,NK+2
         IF (NOD(I,J,K).NE.1) THEN
            DXI    = DXXC(I)
            DYJ    = DYYC(J)
            DZK    = DZZC(K)
            VOL    = DXI*DYJ*DZK
            SUMT   = SUMT+VOL/T(I,J,K)
            SUMPT  = SUMPT+P(I,J,K)*VOL/T(I,J,K)
            SUMPET = SUMPET+REQ(I,J,K)*VOL*(1.-1./T(I,J,K))
         ENDIF
  370 CONTINUE
      IF(NVENT.GT.0) SUMPET =SUMPET+CONVMT
      SUMPET = SUMPET/UGRT
      PCORR  =(SUMPET-SUMPT)/SUMT

      DO 371 I=1,NI+4
      DO 371 J=1,NJ+4
      DO 371 K=1,NK+4
         P(I,J,K) = P(I,J,K)+PCORR
  371 CONTINUE

      IF(NVENT.LT.0) GOTO 373
      DO 372 I=1,2
      DO 372 J=JHWALS,JHWALF
      DO 372 K=KHWALS,KHWALF
         P(I,J,K)=P(I,J,K)-PCORR
  372 CONTINUE
```

```
  373 CONTINUE
      RETURN
      END

************************************************************************
************************************************************************
      SUBROUTINE GRID
************************************************************************
*NONDIMENSIONAL VARIABLES:
*  GRID SIZES:
*    DX = X DIRECTION
*    DY = Y DIRECTION
*    DZ = Z DIRECTION
*
*  CENTRAL CELLS:
*    XC()    = X COORDINATE
*    YC()    = Y COORDINATE
*    ZC()    = Z COORDINATE
*    DXXC()  = X LENGTH
*    DYYC()  = Y LENGTH
*    DZZC()  = Z LENGTH
*
*  STAGGERED CELLS:
*    XS()    = X COORDINATE
*    YS()    = Y COORDINATE
*    ZS()    = Z COORDINATE
*    DXXS()  = X LENGTH
*    DYYS()  = Y LENGTH
*    DZZS()  = Z LENGTH
*
************************************************************************

C *** THIS GRID SHOULD BE CHANGED TO LOCATE FIRST NODES IN
C *** AIR VERY CLOSE TO WALLS

      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      COMMON/R4/XC(40),YC(40),ZC(40),XS(40),YS(40),ZS(40),DXXC(40),
     &          DYYC(40),DZZC(40),DXXS(40),DYYS(40),DZZS(40)
      COMMON/BL1/DX,DY,DZ,DTIME,TCOOL,PI,Q,QR
      COMMON/BL2/X,Y,H,TFLR,TWAL
      COMMON/BL7/NI,NJ,NK,KRUN,NBLOR,NWRP
      COMMON/BL50/JHWALS,JHWALF,KHWALS,KHWALF,NVENT

C *** GENERATION OF THE GRIDS
      DX=X/(DFLOAT(NI-5)*H)
      DY=Y/(DFLOAT(NJ-5)*H)
      DZ=H/(DFLOAT(NK)*H)

C *** CALCULATE XS,YS,ZS (COORDINATES OF STAGGERED CV'S)
      II=0
      L=0
      LL=0
      LLL=0
C ***** X DIRECTION *****
      DO 1 I=3,7
      XS(I)=(I-3)*DX/2.
      L=I
 1    CONTINUE
      DO 2 I=L+1,22
      XS(I)=XS(L)+(I-L)*DX
      LL=I
```

```
  2   CONTINUE
      DO 3 I=LL+1,NI+3
      XS(I)=XS(LL)+(I-LL)*DX/2.
  3   CONTINUE
      XS(2)=XS(3)-TWAL/(H*12.)
      XS(1)=XS(2)-TWAL/(H*12.)
      XS(NI+4)=XS(NI+3)+TWAL/(H*12.)
      XS(NI+5)=XS(NI+4)+TWAL/(H*12.)

C **** Y DIRECTION ****
      DO 6  J=3,4
         YS(J)=(J-3)*DY*2.
 6    CONTINUE
      YS(5)=YS(4)+DY
      YS(6)=YS(5)+DY
      DO 7 J=7,18
        YS(J)=YS(6)+(J-6)*DY/2.
        L=J
 7    CONTINUE
      DO 8 J=L+1,NJ+3
        YS(J)=YS(L)+(J-L)*DY
 8    CONTINUE

      YS(2)=YS(3)-TWAL/(H*12.)
      YS(1)=YS(2)-TWAL/(H*12.)
      YS(NJ+4)=YS(NJ+3)+TWAL/(H*12.)
      YS(NJ+5)=YS(NJ+4)+TWAL/(H*12.)

C ***** Z DIRECTION *****
      DO 10 K=3,NK+3
         ZS(K)=(K-3)*DZ
 10   CONTINUE
      ZS(2)=ZS(3)-TFLR/(H*12.)
      ZS(1)=ZS(2)-TFLR/(H*12.)
      ZS(NK+4)=ZS(NK+3)+TFLR/(H*12.)
      ZS(NK+5)=ZS(NK+4)+TFLR/(H*12.)
      L=0
      LL=0
C *** CALCULATE DXXC,DYYC AND DZZC (DIMENSIONS OF CENTERED CV'S)
      DO 20 I=1,NI+4
         DXXC(I)=XS(I+1)-XS(I)
 20   CONTINUE
      DXXC(NI+5)=DXXC(NI+4)

      DO 22 J=1,NJ+4
         DYYC(J)=YS(J+1)-YS(J)
 22   CONTINUE
      DYYC(NJ+5)=DYYC(NJ+4)

      DO 24 K=1,NK+4
         DZZC(K)=ZS(K+1)-ZS(K)
 24   CONTINUE
      DZZC(NK+5)=DZZC(NK+4)

C *** CALCULATE DXXS,DYYS,DZZS (DIMENSIONS OF STAGGERED CV'S)
      DO 30 I=2,NI+5
         DXXS(I)=(DXXC(I)+DXXC(I-1))/2.0
 30   CONTINUE
      DXXS(1)=DXXS(2)

      DO 32 J=2,NJ+5
```

```
         DYYS(J)=(DYYC(J)+DYYC(J-1))/2.0
 32   CONTINUE
      DYYS(1)=DYYS(2)

      DO 34 K=2,NK+5
         DZZS(K)=(DZZC(K)+DZZC(K-1))/2.0
 34   CONTINUE
      DZZS(1)=DZZS(2)

C *** CALCULATE XC,YC,ZC (LOCATION OF CENTER CELLS)
      DO 40 I=1,NI+5
         XC(I)=XS(I)+DXXC(I)/2.0
 40   CONTINUE

      DO 42 J=1,NJ+5
         YC(J)=YS(J)+DYYC(J)/2.0
 42   CONTINUE

      DO 44 K=1,NK+5
         ZC(K)=ZS(K)+DZZC(K)/2.0
 44   CONTINUE

      RETURN
      END

************************************************************************
************************************************************************
      SUBROUTINE INIT
************************************************************************
*THIS SUBROUTINE INITIALIZES THE FIELD AND CONSTANTS WITH RESPECT
*TO INITIAL START OR RESTARTING CAPABILITY.
*
*VARIABLES ARE :
* ALEW          = LEWIS NUMBER (USED IN SMOKE CONCENTRATION CALCULATIONS)
* BUOY          = BUOYANCY FORCE CONSTANT
* C0            = INITIAL SMOKE CONCENTRATION
* CONDO         = REFERENCE CONDUCTIVITY
* CONSRA        = NONDIMENSIONAL RADIATION CONSTANT
* CPO           = REFERENCE SPECIFIC HEAT
* F             = INITIAL MASS OF FUEL (LBM)
* FR            = MASS OF FUEL REMAINING (LBM)
* GC            = GRAVITY CONSTANT
* H             = CHARACTERISTIC LENGTH;HEIGHT OF CHAMBER=10.FT
* HCOEF         = DIMENSIONLESS HEAT TRANSFER COEF
* HCONV         = HEAT TRANSFER COEFFICIENT IN BTU/(HR*FT**2*DEGREES)
* HR            = HEIGHT IN CM
* NTAPE         = NONDIMENSIONAL FORMS OF TTAPE
* NWRITE        = NONDIMENSIONAL FORMS OF TWRITE
* RHO0          = REFERENCE DENSITY
* TA            = TEMP IN DEGREES RANKINE
* TIME          = DIMENSIONLESS TIME
* TR            = TEMP IN DEGREES KELVIN
* U0            = CHARACTERISTIC VELOCITY (1 FT/SEC)
* UGRT          = PERFECT GAS LAW NONDIMENSIONAL CONSTANT
* VISO          = REFERENCE VISCOSITY (NONDIM)
* VISL          = MINIMUM VISCOSITY (NONDIM)
* VISMAX        = MAXIMUM VISCOSITY (NONDIM)
*
*MATRICES OF THE FORM
* _OD           = DIMENSIONLESS PARAMETER AT PREVIOUS TIME STEP
* _             = DIMENSIONLESS PARAMETER AT CURRENT TIME STEP
```

```
* _PD          = DIMENSIONLESS PARAMETER AT NEXT TIME STEP
*
*WHERE THE PARAMETERS ARE
* AP           = COEFICIENT AT NODE P
* AE,AW,AN     = COEFICIENTS AT PTS EAST,WEST,NORTH,
* AS,AF,AB              SOUTH, FRONT, AND BACK
* CPM          = MEAN SPECIFIC HEAT
* COND( )      = CONDUCTIVITY MATRIX
* CX,CY,CZ     = LOCATION OF THERMOCOUPLE IN X,Y,Z
* DU,DV,DW     = USED IN PRESSURE CORRECTION SUBROUTINE
* DXXC,DYYC    = LENGTH AROUND THE CENTER CELL
* DZZC
* DXXS,DYYS    = LENGTH AROUND THE STAGGERED CELL
* DZZS
* NOD          = IF EQUAL TO ZERO, LIQUID; IF EQUAL TO ONE, SOLID
* PP           = CORRECTED PRESSURE (P')
* REQ          = DENSITY AT EQUILIBRIUM
* SMP          = RESIDUAL MASS SUMMATION OF NODAL POINT
* SMPP         = LENGTH SCALE FOR TURBULENCE
* SP           = BOUNDARY CONDITION TERM AT NODE P
* SU           = SOURCE TERM
* T,P,C        = TEMP, PRESSURE, AND SMOKE CONCENTRATION
* U,V,W        = VELOCITY COMPONENTS IN X,Y,X DIRECTIONS
* VIS          = VISCOSITY
* _B,_E        = BEGINNING AND ENDING NODAL POINT FOR
*                                 THE SOLID IN I,J,K
* XC,YC,ZC     = X,Y,Z LOCATION OF CENTER CELL NODAL POINT
* XS,YS,ZS     = X,Y,Z LOCATION OF STAGGERED CELL NODAL POINT
***********************************************************************

      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      COMMON/R4/XC(40),YC(40),ZC(40),XS(40),YS(40),ZS(40),DXXC(40),
     &          DYYC(40),DZZC(40),DXXS(40),DYYS(40),DZZS(40)
      COMMON/BL1/DX,DY,DZ,DTIME,TCOOL,PI,Q,QR
      COMMON/BL2/X,Y,H,TFLR,TWAL
      COMMON/BL3/F,FR,HSTART
      COMMON/BL7/NI,NJ,NK,KRUN,NBLOR,NWRP
      COMMON/BL12/NWRITE,NTAPE,NTMAX0,NTREAL,TIME,SORSUM,ITER
      COMMON/BL14/HCOEF,CNT,ABTURB,BTURB,VISL,VISMAX
      COMMON/BL16/U0,UGRT,BUOY,CP0,PRT,COND0,VIS0,RHO0,
     &            TA,DTEMP,TWRITE,TTAPE,TMAX,GC,RAIR,NT
      COMMON/BL20/SIG11(30,30,25),SIG12(30,30,25),SIG22(30,30,25),
     &            SIG13(30,30,25),SIG23(30,30,25),SIG33(30,30,25)
      COMMON/BL22/CPS(20),CONS(20),WFAN(20),NCHIP,ICHPB(20),NCHPI(20),
     &            JCHPB(20),NCHPJ(20),KCHPB(20),NCHPK(20)
      COMMON/BL31/TOD(30,30,25),ROD(30,30,25),POD(30,30,25),
     &            COD(30,30,25),UOD(30,30,25),VOD(30,30,25),
     &            WOD(30,30,25)
      COMMON/BL32/T(30,30,25),R(30,30,25),P(30,30,25),C(30,30,25),
     &            U(30,30,25),V(30,30,25),W(30,30,25)
      COMMON/BL33/TPD(30,30,25),RPD(30,30,25),PPD(30,30,25),
     &            CPD(30,30,25),UPD(30,30,25),VPD(30,30,25),
     &            WPD(30,30,25)
      COMMON/BL34/HEIGHT(30,30,25),REQ(30,30,25),SMP(30,30,25),
     &            SMPP(30,30,25),PP(30,30,25),DU(30,30,25),
     &            DV(30,30,25),DW(30,30,25)
      COMMON/BL36/AP(30,30,25),AE(30,30,25),AW(30,30,25),AN(30,30,25),
     &            AS(30,30,25),AF(30,30,25),AB(30,30,25),SP(30,30,25),
     &            SU(30,30,25),RI(30,30,25)
      COMMON/BL37/VIS(30,30,25),COND(30,30,25),RESORM(40),
     &            CPM(30,30,25),NHSZ(3,2),NOD(30,30,25)
```

```
      COMMON/BL38/TCOUP(30),CX(30),CY(30),CZ(30),NTH(30,3),NTHCO
      COMMON/BL39/ALEW,CONSRA,QSIN,QSWER,QSWAL,QSAIR,QSFAN,QSHOL
      COMMON/BL43/QSCONF,QSCONB,QSCONE,QSCONW,QSCONN,QSCONS,QSCONH,
     &            QSRADF, QSRADB,QSRADE,QSRADW,QSRADN,QSRADS,QSRHOL,
     &            WAIR,WWAL,WINS,WERR,WWFAN,WWHOL
      COMMON/BL50/JHWALS,JHWALF,KHWALS,KHWALF,NVENT
      COMMON/BL51/SSGPW(20,20,20)
      COMMON/BL52/CONVMT

C *** INITIALIZE GIVEN PARAMETERS
      C0=0.0
      F=200.0
      HCONV=15.0
      NBLOR=20
      PI=4.*ATAN(1.)
      TCOOL=1.0

C *** NONDIMENSIONALIZE THE REFERENCE VISCOSITY
      VIS0=VIS0/(U0*H)
      VISL=VIS0

C *** SET MAXIMUM VISCOSITY
      VISMAX=400.*VISL

C *** NONDIMENSIONALIZE THE HEAT TRANSFER COEFFICIENT
      HCOEF=HCONV/(3600.*CP0*RHO0*U0)

      COND0=VIS0/PRT
      FR=F
      BUOY=GC*H/(U0**2)
      UGRT=U0**2/(GC*RAIR*TA)
      CONSRA=1.714E-9*TA**3/(RHO0*CP0*U0*3600.)

C *** FOR ENERGY DISTRIBUTION
      QSIN=0.
      QSWER=0.
      QSWAL=0.
      QSAIR=0.
      QSFAN=0.
      QSHOL=0.

C *** INITIALIZE CONDUCTION HEAT FLUX TO EACH WALL
      QSCONF=0.
      QSCONB=0.
      QSCONE=0.
      QSCONW=0.
      QSCONN=0.
      QSCONS=0.
      QSCONH=0.

C *** INITIALIZE RADIATION HEAT FLUX TO EACH WALL
      QSRADF=0.
      QSRADB=0.
      QSRADE=0.
      QSRADW=0.
      QSRADN=0.
      QSRADS=0.
      QSRHOL=0.

      NWRITE=TWRITE*U0/(DTIME*H)
      NTAPE=TTAPE*U0/(DTIME*H)
```

```
      IF(KRUN.LE.0) CONVMT=0.0

C *** INITIALIZE VARIABLE FIELDS
      DO 220 J=1,NJ+4
      DO 220 I=1,NI+4
      DO 220 K=1,NK+4

         IF(KRUN.LE.0) THEN
            UOD(I,J,K)  =0.
            VOD(I,J,K)  =0.
            WOD(I,J,K)  =0.
            POD(I,J,K)  =0.
            TOD(I,J,K)  =TA/TA
            COD(I,J,K)  =C0
         ENDIF

         U(I,J,K)     =UOD(I,J,K)
         UPD(I,J,K)   =UOD(I,J,K)

         V(I,J,K)     =VOD(I,J,K)
         VPD(I,J,K)   =VOD(I,J,K)

         W(I,J,K)     =WOD(I,J,K)
         WPD(I,J,K)   =WOD(I,J,K)

         P(I,J,K)     =POD(I,J,K)
         PPD(I,J,K)   =POD(I,J,K)

         T(I,J,K)     =TOD(I,J,K)
         TPD(I,J,K)   =TOD(I,J,K)

         C(I,J,K)     =COD(I,J,K)
         CPD(I,J,K)   =COD(I,J,K)

         DU(I,J,K)    =0.
         DV(I,J,K)    =0.
         DW(I,J,K)    =0.

         SU(I,J,K)    =0.
         SP(I,J,K)    =0.

         PP(I,J,K)    =0.

         AP(I,J,K)    =0.
         AW(I,J,K)    =0.
         AE(I,J,K)    =0.
         AN(I,J,K)    =0.
         AS(I,J,K)    =0.
         AF(I,J,K)    =0.
         AB(I,J,K)    =0.

         SMP(I,J,K)   =0.
         SMPP(I,J,K)  =0.

         SIG11(I,J,K)=0.
         SIG12(I,J,K)=0.
         SIG13(I,J,K)=0.
         SIG22(I,J,K)=0.
         SIG23(I,J,K)=0.
         SIG33(I,J,K)=0.
```

```
          VIS(I,J,K)   =VISL
          COND(I,J,K)  =COND0
          CPM(I,J,K)   =1.0E0
          NOD(I,J,K)   =0
  220 CONTINUE
       DO 221 I=1,20
       DO 221 J=1,20
       DO 221 K=1,20
        SSGPW(I,J,K)=0.0
 221   CONTINUE

C *** DEFINE THERMAL PROPERTIES OF DECK AND SOLID
      IF (NCHIP.NE.0) CALL SOLCON

C *** DEFINE HEIGHT OF NODE POINTS AND COMPUTE HYDROSTATIC
C     EQUILIBRIUM DENSITY REQ(I,J,K)
   15 DO 229 J=1,NJ+4
      DO 229 I=1,NI+4
      DO 229 K=1,NK+4
         HEIGHT(I,J,K)=ZC(K)
         REQ(I,J,K)   =EXP(-BUOY*UGRT*HEIGHT(I,J,K))
         IF(KRUN.LE.0) THEN
            ROD(I,J,K)=REQ(I,J,K)/TPD(I,J,K)
         ENDIF
         R(I,J,K)   =ROD(I,J,K)
         RPD(I,J,K)=ROD(I,J,K)
  229 CONTINUE

C *** FOLLOWING IS FOR DETERMINING THE THERMOCOUPLE POSITIONS
      DO 5000 N=1,NTHCO
         DO 5001 I=1,NI+4
            IF (XC(I).LT.CX(N).AND.XC(I+1).GE.CX(N)) GOTO 5002
 5001    CONTINUE

 5002    II=I
         DO 5003 J=1,NJ+4
            IF (YC(J).LT.CY(N).AND.YC(J+1).GE.CY(N)) GOTO 5004
 5003    CONTINUE

 5004    JJ=J
         DO 5005 K=1,NK+4
            IF (ZC(K).LT.CZ(N).AND.ZC(K+1).GE.CZ(N)) GOTO 5006
 5005    CONTINUE

 5006    KK=K
         NTH(N,1)=II
         NTH(N,2)=JJ
         NTH(N,3)=KK
 5000 CONTINUE

      RETURN
      END

************************************************************************
************************************************************************
      SUBROUTINE INPUT(NSTOP)
************************************************************************
*THIS SUBROUTINE SETS UP REQUIRED VALUES TO BEGIN THE PROGRAM.
*
*VARIABLES ARE:
```

```
*  KRUN        = RESTART INDICATOR
*  NCHIP       = NUMBER OF INTERNAL SOLID PIECES
*  NMS         = NUMBER OF MASS SOURCES
*  NWRP        = NUMBER OF TIME STEPS BETWEEN WRITES TO OUTPUT FILE
*  NTHCO       = NUMBER OF THERMOCOUPLES TO PRINT OUT
*  TMAX        = NONDIMENSIONAL MAXIMUM TIME ALLOWED
*  XTMAX       = MAXIMUM TIME ALLOWED (SECONDS)
*  TWRITE      = TIME BETWEEN FIELD VARIABLE OUTPUT (SECONDS)
*  TTAPE       = TIME INTERVAL BETWEEN PLOTS (SECONDS)
*  DTIME       = NONDIMENSIONAL TIME STEP
*  XDTIME      = TIME STEP (SECONDS)
*  HSTART      = FIRE START TIME (SECONDS)
*  NHSZ(1,1)  = STARTING NODE OF HEAT SOURCE, X-DIR
*  NHSZ(2,1)  =                               Y-DIR
*  NHSZ(3,1)  =                               Z-DIR
*  NHSZ(1,2)  = ENDING NODE OF HEAT SOURCE, X-DIR
*  NHSZ(2,2)  =                             Y-DIR
*  NHSZ(3,2)  =                             Z-DIR
*  ICHPB       = FIRST NODE OF INTERNAL SOLID IN X DIR
*  JCHPB       =                                 Y DIR
*  KCHPB       =                                 Z DIR
*  NCHPI       = NUMBER OF INTERNAL SOLID NODES IN X DIR
*  NCHPJ       =                                   Y DIR
*  NCHPK       =                                   Z DIR
*  IMSB        = FIRST MASS SOURCE NODE IN X DIR
*  JMSB        =                           Y DIR
*  KMSB        =                           Z DIR
*  NMSI        = NUMBER OF MASS SOURCE NODES IN X DIR
*  NMSJ        =                                Y DIR
*  NMSK        =                                Z DIR
*  RMS         = DIMENSIONLESS MASS SOURCE
*                (= CFM/(60.*H**2*U0*NMSI*NMSJ*NMSK)
*  CX,CY,CZ   = THERMOCOUPLE POSITIONS IN X,Y,Z
************************************************************************
*DATA FILES USED IN THIS PROGRAM:
*
* FILE # 10 = FIRE.DAT   : INITIAL SET-UP DATA
*        11 = FIRE1.CONT : RESTART/CONTINUATION DATA
*
************************************************************************

      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      COMMON/BL1/DX,DY,DZ,DTIME,TCOOL,PI,Q,QR
      COMMON/BL2/X,Y,H,TFLR,TWAL
      COMMON/BL3/F,FR,HSTART
      COMMON/BL7/NI,NJ,NK,KRUN,NBLOR,NWRP
      COMMON/BL12/NWRITE,NTAPE,NTMAX0,NTREAL,TIME,SORSUM,ITER
      COMMON/BL16/U0,UGRT,BUOY,CP0,PRT,COND0,VIS0,RHO0,
     &            TA,DTEMP,TWRITE,TTAPE,TMAX,GC,RAIR,NT
      COMMON/BL23/RMS(20),NMS,IMSB(20),NMSI(20),JMSB(20),NMSJ(20),
     &            KMSB(20),NMSK(20)
      COMMON/BL22/CPS(20),CONS(20),WFAN(20),NCHIP,ICHPB(20),NCHPI(20),
     &            JCHPB(20),NCHPJ(20),KCHPB(20),NCHPK(20)
      COMMON/BL31/TOD(30,30,25),ROD(30,30,25),POD(30,30,25),
     &            COD(30,30,25),UOD(30,30,25),VOD(30,30,25),
     &            WOD(30,30,25)
      COMMON/BL37/VIS(30,30,25),COND(30,30,25),RESORM(40),
     &            CPM(30,30,25),NHSZ(3,2),NOD(30,30,25)
      COMMON/BL38/TCOUP(30),CX(30),CY(30),CZ(30),NTH(30,3),NTHCO
      COMMON/BL50/JHWALS,JHWALF,KHWALS,KHWALF,NVENT
      COMMON/BL52/CONVMT
```

```
      CHARACTER ANS*1
      LOGICAL L1,L2
      NSTOP=0
      KRUN=0

***** CHECK FOR INPUT DATA FILE
      INQUIRE (FILE='/FIRE DATA B1',EXIST=L1)
      IF (L1) THEN

C *** READ IN DATA FROM EXISTING DATA FILE
         OPEN(10,FILE='/FIRE DATA B1',STATUS='OLD')
         REWIND(10)
         READ(10,*) X,Y,H,TFLR,TWAL,TA
         READ(10,*) NI,NJ,NK
         READ(10,*) NCHIP,NMS,NWRP,NTHCO
         READ(10,*) TMAX,DTIME,TTAPE,TWRITE,HSTART
         READ(10,*) NHSZ(1,1),NHSZ(1,2),NHSZ(2,1),NHSZ(2,2),NHSZ(3,1),
     &              NHSZ(3,2)
         IF (NCHIP.LE.0) GOTO 33
         DO 32 N=1,NCHIP
            READ(10,*) ICHPB(N),NCHPI(N),JCHPB(N),NCHPJ(N),KCHPB(N),
     &                 NCHPK(N),CPS(N),CONS(N),WFAN(N)
   32    CONTINUE
   33    IF (NMS.LE.0) GOTO 37
         DO 36 N=1,NMS
            READ(10,*) IMSB(N),NMSI(N),JMSB(N),NMSJ(N),KMSB(N),
     &                 NMSK(N),RMS(N)
   36    CONTINUE
   37    DO 38 I=1,NTHCO
            READ (10,*) CX(I),CY(I),CZ(I)
   38    CONTINUE
         READ(10,*) JHWALS,JHWALF,KHWALS,KHWALF,NVENT
         REWIND(10)
         CLOSE(10)
      ELSE

C *** STOP PROGRAM IF INPUT DATA NOT AVAILABLE
         NSTOP=9999
         GOTO 999
      ENDIF

***** CHECK FOR CONTINUATION FILE
      INQUIRE (FILE='/CONTINUE DATA B4',EXIST=L2)
      IF (L2) THEN

C *** READ IN DATA FROM OLD CONTINUATION FILE
         OPEN(11,FILE='/CONTINUE DATA B4',STATUS='OLD',
     &           FORM='UNFORMATTED')
         KRUN=1
         REWIND(11)
         READ(11) TIME,NTMAX0,CONVMT,FR,TOD,ROD,UOD,VOD,WOD,POD,COD
         REWIND(11)
         IF(TIME.GE.TMAX) TMAX=TIME+TMAX
      ELSE

C *** CREATE NEW CONTINUATION FILE
         OPEN(11,FILE='/CONTINUE DATA B4',STATUS='NEW',
     &           FORM='UNFORMATTED')
         KRUN=0
      ENDIF
```

```
  999 RETURN
      END

***********************************************************************
***********************************************************************
      SUBROUTINE OUT(NN)
***********************************************************************
*THIS SUBROUTINE GENERATES OUTPUT.
*
* NN = 1      SELECTED VALUES ARE PRINTED. INCLUDING TIME, ERROR,
*             PRESSURE, HEAT GENERATION
* NN = 2      TEMPERATURE AT THE THERMOCOUPLES
* NN = 3      FILED VALUES ARE PRINTED
* NN = 4      ENERGY DISTRIBUTION
***********************************************************************

      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      COMMON/BL1/DX,DY,DZ,DTIME,TCOOL,PI,Q,QR
      COMMON/BL2/X,Y,H,TFLR,TWAL
      COMMON/BL7/NI,NJ,NK,KRUN,NBLOR,NWRP
      COMMON/BL12/NWRITE,NTAPE,NTMAX0,NTREAL,TIME,SORSUM,ITER
      COMMON/BL16/U0,UGRT,BUOY,CP0,PRT,COND0,VIS0,RHO0,
     &            TA,DTEMP,TWRITE,TTAPE,TMAX,GC,RAIR,NT
      COMMON/BL32/T(30,30,25),R(30,30,25),P(30,30,25),C(30,30,25),
     &            U(30,30,25),V(30,30,25),W(30,30,25)
      COMMON/BL34/HEIGHT(30,30,25),REQ(30,30,25),SMP(30,30,25),
     &            SMPP(30,30,25),PP(30,30,25),DU(30,30,25),
     &            DV(30,30,25),DW(30,30,25)
      COMMON/BL36/AP(30,30,25),AE(30,30,25),AW(30,30,25),AN(30,30,25),
     &            AS(30,30,25),AF(30,30,25),AB(30,30,25),SP(30,30,25),
     &            SU(30,30,25),RI(30,30,25)
      COMMON/BL37/VIS(30,30,25),COND(30,30,25),RESORM(40),
     &            CPM(30,30,25),NHSZ(3,2),NOD(30,30,25)
      COMMON/BL38/TCOUP(30),CX(30),CY(30),CZ(30),NTH(30,3),NTHCO
      COMMON/BL39/ALEW,CONSRA,QSIN,QSWER,QSWAL,QSAIR,QSFAN,QSHOL
      COMMON/BL43/QSCONF,QSCONB,QSCONE,QSCONW,QSCONN,QSCONS,QSCONH,
     &            QSRADF, QSRADB,QSRADE,QSRADW,QSRADN,QSRADS,QSRHOL,
     &            WAIR,WWAL,WINS,WERR,WWFAN,WWHOL
      COMMON/BL50/JHWALS,JHWALF,KHWALS,KHWALF,NVENT
      COMMON/BL52/CONVMT

***********************************************************************

  500 FORMAT(1X,'TIME=',F7.3,' SECONDS',2X,'NTREAL=',I5,2X,'ITER=',I5,
     &       2X,'SOURCE=',F9.6,2X,'SORSUM=',F9.6,2X,' Q(KW) = ',F10.4)
  504 FORMAT(/,1X,'I=',I2,5X,'J=',I2,/,10X,'T (C)',
     &        4X,'VIS',10X,'COND',8X,'DENS',/)
C    &        4X,'U(CM/SEC)',2X,'V(CM/SEC)',2X,'W(CM/SEC)',/)
  511 FORMAT(1X,'K=',I3,2X,E10.3,2X,E10.3,2X,E10.3,2X,E10.3)
  560 FORMAT(8X,A4,10X,A4,12X,A4,12X,A4)
  561 FORMAT(4(4X,E12.5))
 1084 FORMAT(6X,'*** AT TIME = ',F9.4,' ***',/,
     &       9X,'THE WATTAGE INPUT: WINS = ',E11.4,/,
     &       9X,'WATTAGE INTO AIR:  WAIR = ',E11.4,/,
     &       9X,'WATTAGE DUE TO ERRO     = ',E11.4,/,
     &       9X,'WATTAGE THRU DUCT: WAIR = ',E11.4,/,
     &       9X,'INTO THE WALL:      WWAL = ',E11.4,/,
     &       9X,'INTO THE HOLE:     WWHOL = ',E11.4,/,
     &       2X,'BY CONDUCTION: QTCON = ',E11.4,/,
     &       2X,'BY RADIATION:  QTRAD = ',E11.4,/,
```

```
     &        2X,'BY CONVECTION:QTCONV = ',E11.4,//)
 2728 FORMAT(9X,'QSRADW: INTO WEST  WALL BY RADIATION= ',E11.4,/,
     &        9X,'QSCONW: BY CONDUCTION = ',E11.4,/,
     &        9X,'QSRADE: INTO EAST  WALL BY RADIATION= ',E11.4,/,
     &        9X,'QSCONE: BY CONDUCTION = ',E11.4,/,
     &        9X,'QSRADS: INTO SOUTH WALL BY RADIATION= ',E11.4,/,
     &        9X,'QSCONS: BY CONDUCTION = ',E11.4,/,
     &        9X,'QSRADN: INTO NORTH WALL BY RADIATION= ',E11.4,/,
     &        9X,'QSCONN: BY CONDUCTION = ',E11.4,/,
     &        9X,'QSRADB: INTO BACK  WALL BY RADIATION= ',E11.4,/,
     &        9X,'QSCONB: BY CONDUCTION = ',E11.4,/,
     &        9X,'QSRADF: INTO FRONT WALL BY RADIATION= ',E11.4,/,
     &        9X,'QSCONF: BY CONDUCTION = ',E11.4,/,
     &        9X,'QSRHOL: INTO THE HOLE   BY RADIATION= ',E11.4,/,
     &        9X,'QSCONH: BY CONVECTION = ',E11.4,/)
 1088 FORMAT(9X,'PAIR : LOSS INTO CAVITY AIR  = ',F8.3,' %',/,
     &        9X,'PWER : LOSS DUE TO THE ERR   = ',F8.3,' %',/,
     &        9X,'PWALL: LOSS INTO THE WALLS   = ',F8.3,' %',/,
     &        9X,'PFAN : LOSS THROUGH DUCT     = ',F8.3,' %',/,
     &        9X,'PHOL : LOSS THROUGH HOLE     = ',F8.3,' %',/,
     &        9X,'PSAIR: TOTAL INTO CAVITY AIR = ',F8.3,' %',/,
     &        9X,'PSWER: TOTAL DUE TO THE ERRO = ',F8.3,' %',/,
     &        9X,'PSFAN: TOTAL THROUGH DUCT    = ',F8.3,' %',/,
     &        9X,'PSWAL: TOTAL INTO THE WALLS  = ',F8.3,' %',/,
     &        9X,'PSHOL: TOTAL INTO THE HOLE   = ',F8.3,' %',//)
 1091 FORMAT(9X,'QSIN  : TOTAL ENERGY INPUT            = ',E11.4,/,
     &        9X,'QSWER : TOTAL ENERGY DUE TO THE ERRO= ',E11.4,/,
     &        9X,'QSAIR : TOTAL ENERGY INTO CAVITY AIR= ',E11.4,/,
     &        9X,'QSFAN : TOTAL ENERGY THROUGH DUCT   = ',E11.4,/,
     &        9X,'QSWAL : TOTAL ENERGY INTO WALLS     = ',E11.4,/,
     &        9X,'QSHOL : TOTAL ENERGY INTO HOLE      = ',E11.4,2X,//)
 1095 FORMAT(9X,'MASS FLOW RATE THROUGH HOLE (LBM/S) = ',E11.4,//)
************************************************************************

C *** REFERENCE TEMPERATURE IN DEGREES K
      TR=TA/1.8

C *** REFERENCE VELOCITY IN CM/SEC
      UR=U0*30.48

C *** REFERENCE LENGTH IN CM
      HR=H*30.48

      XTIME=TIME*H/U0
      IF (NN.EQ.1) THEN
         QRR=60.**2*QR/3412.
         QKW=60.**2*Q /3412.
         WRITE(12,500) XTIME,NTREAL,ITER,RESORM(ITER),SORSUM,QKW
      ELSE IF (NN.EQ.2) THEN
         WRITE (12,*)
         WRITE (12,*)' TEMPERATURES  AT THERMOCOUPLE POSITION IN (C):',
     &                 (TCOUP(N)*TR-273.16,N=1,NTHCO)
         WRITE (12,*)
         WRITE (12,*)
      ELSE IF (NN.EQ.3) THEN
         WRITE(12,'(1X,A,F10.6)') 'TIME =',XTIME
         DO 501 I=4,26,11
            DO 502 J=6,26,20
               WRITE(12,504) I,J
               DO 503 K=5,19,7
                  IF (T(I,J,K).LT.TCOOL) T(I,J,K)=TCOOL
```

```
                 XTEMP=T(I,J,K)*TR-273.16
                 XR   =1000.*(0.0328)**3*R(I,J,K)*RHO0/2.2048
C                XU   =U(I,J,K)*UR
C                XV   =V(I,J,K)*UR
C                XW   =W(I,J,K)*UR
C                XP   =P(I,J,K)*RHO0*U0**2/(GC*14.696*144.)+REQ(I,J,K)
                 XVIS =VIS(I,J,K)*HR*UR
                 XCOND=COND(I,J,K)*HR*UR
                 WRITE(12,511)K,XTEMP,XVIS,XCOND,XR
  503          CONTINUE
  502        CONTINUE
  501      CONTINUE


      WRITE(12,560) 'XP1','XP2','XP3','XP4'

      DO 550 K=3,23,4
      XP1  =P(4,22,K)*RHO0*U0**2/(GC*14.696*144.)+REQ(4,22,K)
      XP2  =P(6,6,K)*RHO0*U0**2/(GC*14.696*144.)+REQ(6,6,K)
      XP3  =P(20,12,K)*RHO0*U0**2/(GC*14.696*144.)+REQ(20,12,K)
      XP4  =P(15,18,K)*RHO0*U0**2/(GC*14.696*144.)+REQ(15,18,K)
      WRITE(12,561) XP1,XP2,XP3,XP4
  550 CONTINUE

C **** FIND MAX TEMPS ****
C WEST
       TRTMAX=0.0
      DO 530 I=1,2
      DO 530 J=1,NJ+4
      DO 530 K=1,NK+4
          IF ((J.LE.JHWALS-1 .OR. J.GE.JHWALF+1).OR.
     &         (K.LE.KHWALS-1 .OR. K.GE.KHWALF+1)) THEN

              IF(T(I,J,K).GT.TRTMAX) TRTMAX=T(I,J,K)
          END IF
  530  CONTINUE
            WRITE(12,*) 'WEST WALL MAX TEMP =',TRTMAX*TR-273.16

C EAST
       TRTMAX=0.0
            DO 531 I=28,29
            DO 531 J=1,NJ+4
            DO 531 K=1,NK+4
             IF(T(I,J,K).GT.TRTMAX) TRTMAX=T(I,J,K)
  531  CONTINUE
            WRITE(12,*) 'EAST WALL MAX TEMP =',TRTMAX*TR-273.16
C TOP
       TRTMAX=0.0
            DO 532 K=23,24
            DO 532 J=1,NJ+4
            DO 532 I=1,NI+4
             IF(T(I,J,K).GT.TRTMAX) TRTMAX=T(I,J,K)
  532  CONTINUE
            WRITE(12,*) 'TOP  WALL MAX TEMP =',TRTMAX*TR-273.16
C BOT
       TRTMAX=0.0
            DO 533 K=1,2
            DO 533 J=1,NJ+4
            DO 533 I=1,NI+4
             IF(T(I,J,K).GT.TRTMAX) TRTMAX=T(I,J,K)
  533  CONTINUE
```

```
            WRITE(12,*) 'BOT  WALL MAX TEMP =',TRTMAX*TR-273.16
C SOUTH
        TRTMAX=0.0
            DO 534 J=1,2
            DO 534 K=1,NK+4
            DO 534 I=1,NI+4
             IF(T(I,J,K).GT.TRTMAX) TRTMAX=T(I,J,K)
  534   CONTINUE
            WRITE(12,*) 'SOUTH WALL MAX TEMP =',TRTMAX*TR-273.16
C SOUTH
        TRTMAX=0.0
            DO 535 J=28,29
            DO 535 K=1,NK+4
            DO 535 I=1,NI+4
             IF(T(I,J,K).GT.TRTMAX) TRTMAX=T(I,J,K)
  535   CONTINUE
            WRITE(12,*) 'NORTH WALL MAX TEMP =',TRTMAX*TR-273.16
      ELSE

C *** CALCULATE THE PERCENTAGE AND PRINT OUT THE RESULTS
        QTCON=QSCONF+QSCONB+QSCONS+QSCONN+QSCONW+QSCONE
        QTRAD=QSRADF+QSRADB+QSRADS+QSRADN+QSRADW+QSRADE+QSRHOL
        QTCONV=QSCONH

C *** WATT PERCENTAGE
        IF (WINS.EQ.0.) WINS=1.0E-5
        PAIR=100.*WAIR /WINS
        PWAL=100.*WWAL /WINS
        PFAN=100.*WWFAN/WINS
        PWER=100.*WERR /WINS
        PHOL=100.*WWHOL/WINS

C *** ENERGY PERCENTAGE
        IF (QSIN.EQ.0.0) QSIN=1.0E-3
        PSAIR=100.*QSAIR/QSIN
        PSWAL=100.*QSWAL/QSIN
        PSFAN=100.*QSFAN/QSIN
        PSWER=100.*QSWER/QSIN
        PSHOL=100.*QSHOL/QSIN
        WRITE(12,1084)XTIME,WINS,WAIR,WERR,WWFAN,WWAL,WWHOL,
     &               QTCON,QTRAD,QTCONV
        WRITE(12,2728)QSRADW,QSCONW,QSRADE,QSCONE,QSRADS,QSCONS,
     &               QSRADN,QSCONN,QSRADB,QSCONB,QSRADF,QSCONF,
     &               QSRHOL,QSCONH
        WRITE(12,1088)PAIR,PWER,PWAL,PFAN,PHOL,PSAIR,PSWER,PSFAN,
     &               PSWAL,PSHOL
        WRITE(12,1091)QSIN,QSWER,QSAIR,QSFAN,QSWAL,QSHOL
        WRITE(12,1095) CONVMT*RHO0*H**2
      ENDIF

      RETURN
      END
************************************************************************
************************************************************************
      SUBROUTINE RADHT (NN)
************************************************************************
*  NN = CONTROL PARAMETER FOR HEAT FLUX CALCULATIONS
*       WHERE NN=1 :CALCULATE HEAT FLUX FROM FIRE TO WALLS
*             NN=2 :CALCULATE HEAT FLUX FROM FIRE TO BLOCKS
*
*  NTHS        = TOTAL NUMBER OF CV'S CONTAINING HEAT SOURCES
```

```
*  NFX,NFY,NFZ = CV NUMBER OF HEAT SOURCE
*  FX,FY,FZ    = COORDINATES OF HEAT SOURCE
*************************************************************************

      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      COMMON/R4/XC(40),YC(40),ZC(40),XS(40),YS(40),ZS(40),DXXC(40),
     &          DYYC(40),DZZC(40),DXXS(40),DYYS(40),DZZS(40)
      COMMON/BL1/DX,DY,DZ,DTIME,TCOOL,PI,Q,QR
      COMMON/BL7/NI,NJ,NK,KRUN,NBLOR,NWRP
      COMMON/BL22/CPS(20),CONS(20),WFAN(20),NCHIP,ICHPB(20),NCHPI(20),
     &            JCHPB(20),NCHPJ(20),KCHPB(20),NCHPK(20)
      COMMON/BL32/T(30,30,25),R(30,30,25),P(30,30,25),C(30,30,25),
     &            U(30,30,25),V(30,30,25),W(30,30,25)
      COMMON/BL36/AP(30,30,25),AE(30,30,25),AW(30,30,25),AN(30,30,25),
     &            AS(30,30,25),AF(30,30,25),AB(30,30,25),SP(30,30,25),
     &            SU(30,30,25),RI(30,30,25)
      COMMON/BL37/VIS(30,30,25),COND(30,30,25),RESORM(40),
     &            CPM(30,30,25),NHSZ(3,2),NOD(30,30,25)
      COMMON/BL39/ALEW,CONSRA,QSIN,QSWER,QSWAL,QSAIR,QSFAN,QSHOL
      COMMON/BL40/VFHSW(8,30,30),VFHSE(8,30,30),VFHSS(8,30,30),
     &            VFHSN(8,30,30),VFHSB(8,30,30),VFHSF(8,30,30)
      COMMON/BL41/VFHSBW(5,8,34,34),VFHSBE(5,8,34,34),VFHSBS(5,8,34,34),
     &            VFHSBN(5,8,34,34),VFHSBB(5,8,34,34),VFHSBF(5,8,34,34)
      COMMON/BL50/JHWALS,JHWALF,KHWALS,KHWALF,NVENT
      COMMON/BL51/SSGPW(20,20,20)

      NTHS=NHSZ(3,2)-NHSZ(3,1)+1
      DO 500 N=1,NTHS
         NFX=NHSZ(1,1)
         NFY=NHSZ(2,1)
         NFZ=NHSZ(3,1)-1+N

C *** AREA OF THE FIRE ELEMENTS
         DYHS=DYYC(NFY)
         DZHS=DZZC(NFZ)
         DAHS=PI*DYHS*DZHS
         EMIS=0.6
         EMISS1=1.0

C *** NN=1:  CALCULATE RADIATION HEAT FLUX FROM FIRE TO WEST AND
C ***        EAST SURFACES OF THE ENCLOSURE
         IF (NN.EQ.1) THEN
            DO 100 J=3,NJ+2
            DO 100 K=3,NK+2
            IF(NVENT.GT.0) THEN
             IF((K.LE.KHWALS-1 .OR. K.GE.KHWALF+1).OR.
     &          (J.LE.JHWALS-1 .OR. J.GE.JHWALF+1)) THEN
               SU(2,J,K) = SU(2,J,K)+ CONSRA*EMIS*DAHS*VFHSW(N,J,K)*
     &                     (T(NFX,NFY,NFZ)**4-T(2,J,K)**4)
             END IF
            ELSE
               SU(2,J,K) = SU(2,J,K)+ CONSRA*EMIS*DAHS*VFHSW(N,J,K)*
     &                     (T(NFX,NFY,NFZ)**4-T(2,J,K)**4)
            END IF
               SU(NI+3,J,K)=SU(NI+3,J,K)+CONSRA*EMIS*DAHS*VFHSE(N,J,K)*
     &                        (T(NFX,NFY,NFZ)**4-T(NI+3,J,K)**4)

  100       CONTINUE

            IF(NVENT.LT.0) GOTO 598
            DO 110 J=JHWALS,JHWALF
```

```
          DO 110 K=KHWALS,KHWALF
             SSGPW(2,J,K)=CONSRA*EMISS1*DAHS*VFHSW(N,J,K)*
     &                    (T(NFX,NFY,NFZ)**4-TCOOL**4)
 110      CONTINUE
 598      CONTINUE

C *** CALCULATE RADIATION HEAT FLUX FROM FIRE TO NORTH AND
C *** SOUTH SURFACES OF THE ENCLOSURE
          DO 200 I=3,NI+2
          DO 200 K=3,NK+2
             SU(I,2,K)    =SU(I,2,K)+CONSRA*EMIS*DAHS*VFHSS(N,K,I)*
     &                    (T(NFX,NFY,NFZ)**4-T(I,2,K)**4)
             SU(I,NJ+3,K)=SU(I,NJ+3,K)+CONSRA*EMIS*DAHS*VFHSN(N,K,I)*
     &                    (T(NFX,NFY,NFZ)**4-T(I,NJ+3,K)**4)
 200      CONTINUE

C *** CALCULATE RADIATION HEAT FLUX FROM FIRE TO BACK AND
C *** FRONT SURFACES OF THE ENCLOSURE
          DO 300 I=3,NI+2
          DO 300 J=3,NJ+2
             SU(I,J,2)    =SU(I,J,2)+CONSRA*EMIS*DAHS*VFHSB(N,I,J)*
     &                    (T(NFX,NFY,NFZ)**4-T(I,J,2)**4)
             SU(I,J,NK+3)=SU(I,J,NK+3)+CONSRA*EMIS*DAHS*VFHSF(N,I,J)*
     &                    (T(NFX,NFY,NFZ)**4-T(I,J,NK+3)**4)
 300      CONTINUE
       ENDIF

C *** NN=2:  CALCULATE RADIATION HEAT FLUX FROM FIRE TO WEST AND
C ***        EAST SURFACES OF BLOCK M
       IF (NN.EQ.2) THEN
          IF (NCHIP.GT.NBLOR-1) THEN
             DO 900 M=1,NCHIP-NBLOR+1
                IB=ICHPB(M+NBLOR-1)
                IE=IB+NCHPI(M+NBLOR-1)-1
                JB=JCHPB(M+NBLOR-1)
                JE=JB+NCHPJ(M+NBLOR-1)-1
                KB=KCHPB(M+NBLOR-1)
                KE=KB+NCHPK(M+NBLOR-1)-1

C *** CALCULATE RADIATION HEAT FLUX FROM FIRE TO WEST AND
C *** EAST SURFACES OF THE BLOCK
                DO 400 J=JB,JE-1
                DO 400 K=KB,KE-1
                   SU(IB,J,K)   =SU(IB,J,K)+CONSRA*EMIS*DAHS*
     &                           VFHSBW(N,M,J,K)*(T(NFX,NFY,NFZ)**4-
     &                           T(IB,J,K)**4)
                   SU(IE-1,J,K)=SU(IE-1,J,K)+CONSRA*EMIS*DAHS*
     &                           VFHSBE(N,M,J,K)*(T(NFX,NFY,NFZ)**4-
     &                           T(IE-1,J,K)**4)
 400            CONTINUE

C *** CALCULATE RADIATION HEAT FLUX FROM FIRE TO NORTH AND
C *** SOUTH SURFACES OF THE BLOCK
                DO 600 I=IB,IE-1
                DO 600 K=KB,KE-1
                   SU(I,JB,K)   =SU(I,JB,K)+CONSRA*EMIS*DAHS*
     &                           VFHSBS(N,M,K,I)*(T(NFX,NFY,NFZ)**4-
     &                           T(I,JB,K)**4)
                   SU(I,JE-1,K)=SU(I,JE-1,K)+CONSRA*EMIS*DAHS*
     &                           VFHSBN(N,M,K,I)*(T(NFX,NFY,NFZ)**4-
     &                           T(I,JE-1,K)**4)
```

```
  600              CONTINUE

C *** CALCULATE RADIATION HEAT FLUX FROM FIRE TO BACK AND
C *** FRONT SURFACES OF BLOCK
                   DO 700 I=IB,IE-1
                   DO 700 J=JB,JE-1
                      SU(I,J,KB)   =SU(I,J,KB)+CONSRA*EMIS*DAHS*
     &                              VFHSBB(N,M,I,J)*(T(NFX,NFY,NFZ)**4-
     &                              T(I,J,KB)**4)
                      SU(I,J,KE-1)=SU(I,J,KE-1)+CONSRA*EMIS*DAHS*
     &                              VFHSBF(N,M,I,J)*(T(NFX,NFY,NFZ)**4-
     &                              T(I,J,KE-1)**4)
  700              CONTINUE
  900           CONTINUE
             ENDIF
          ENDIF
  500 CONTINUE

      RETURN
      END

***********************************************************************
***********************************************************************
      SUBROUTINE SOLCON
***********************************************************************
*   THIS SUBROUTINE RESETS THE CONDUCTIVITY OF THE SOLID. IN CALVIS    *
*   THE VISCOSITY ARE CALCULATED  AT EVERY CELL INCLUDING THOSE        *
*   CONTAINING SOLID ONES.                                             *
***********************************************************************

      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      COMMON/BL2/X,Y,H,TFLR,TWAL
      COMMON/BL7/NI,NJ,NK,KRUN,NBLOR,NWRP
      COMMON/BL16/U0,UGRT,BUOY,CP0,PRT,COND0,VIS0,RHO0,
     &            TA,DTEMP,TWRITE,TTAPE,TMAX,GC,RAIR,NT
      COMMON/BL22/CPS(20),CONS(20),WFAN(20),NCHIP,ICHPB(20),NCHPI(20),
     &            JCHPB(20),NCHPJ(20),KCHPB(20),NCHPK(20)
      COMMON/BL37/VIS(30,30,25),COND(30,30,25),RESORM(40),
     &            CPM(30,30,25),NHSZ(3,2),NOD(30,30,25)


      DO 402 N=1,NCHIP
         IB=ICHPB(N)
         IE=IB+NCHPI(N)-1
         JB=JCHPB(N)
         JE=JB+NCHPJ(N)-1
         KB=KCHPB(N)
         KE=KB+NCHPK(N)-1
         DO 405 I=IB,IE-1
         DO 405 J=JB,JE-1
         DO 405 K=KB,KE-1
            COND(I,J,K)=CONS(N)*COND0
            CPM(I,J,K)=CPS(N)
            NOD(I,J,K)=1

C *** SET VALUE AT CORNER OR BOUNDARIES FOR BOUNDARY CONDITIONS
            IF (I.EQ.2) THEN
               COND(1,J,K)=COND(2,J,K)
               CPM (1,J,K)=CPM (2,J,K)
            ELSEIF (I.EQ.NI+3) THEN
```

```
            COND(NI+4,J,K)=COND(NI+3,J,K)
            CPM (NI+4,J,K)=CPM (NI+3,J,K)
         ENDIF

         IF (J.EQ.2) THEN
            COND(I,1,K)=COND(I,2,K)
            CPM (I,1,K)=CPM (I,2,K)
         ELSEIF (J.EQ.NJ+3) THEN
            COND(I,NJ+4,K)=COND(I,NJ+3,K)
            CPM (I,NJ+4,K)=CPM (I,NJ+3,K)
         ENDIF

         IF (K.EQ.2) THEN
            COND(I,J,1)=COND(I,J,2)
            CPM (I,J,1)=CPM (I,J,2)
         ELSEIF (K.EQ.NK+3) THEN
            COND(I,J,NK+4)=COND(I,J,NK+3)
            CPM (I,J,NK+4)=CPM (I,J,NK+3)
         ENDIF

 405    CONTINUE
 402 CONTINUE
      RETURN
      END

***********************************************************************
***********************************************************************
      SUBROUTINE STRESS
***********************************************************************
*THIS SUBROUTINE CALCULATES THE SHEAR STRESSES

      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      COMMON/R4/XC(40),YC(40),ZC(40),XS(40),YS(40),ZS(40),DXXC(40),
     &          DYYC(40),DZZC(40),DXXS(40),DYYS(40),DZZS(40)
      COMMON/BL7/NI,NJ,NK,KRUN,NBLOR,NWRP
      COMMON/BL20/SIG11(30,30,25),SIG12(30,30,25),SIG22(30,30,25),
     &            SIG13(30,30,25),SIG23(30,30,25),SIG33(30,30,25)
      COMMON/BL32/T(30,30,25),R(30,30,25),P(30,30,25),C(30,30,25),
     &            U(30,30,25),V(30,30,25),W(30,30,25)
      COMMON/BL37/VIS(30,30,25),COND(30,30,25),RESORM(40),
     &            CPM(30,30,25),NHSZ(3,2),NOD(30,30,25)

      DO 100 K=2,NK+3
      DO 100 J=2,NJ+3
      DO 100 I=2,NI+3

C *** CENTRAL LENGTH OF THE SCALAR CONTROL VOLUME
         DXP1=DXXC(I+1)
         DXI =DXXC(I)
         DXM1=DXXC(I-1)

         DYP1=DYYC(J+1)
         DYJ =DYYC(J)
         DYM1=DYYC(J-1)

         DZP1=DZZC(K+1)
         DZK =DZZC(K)
         DZM1=DZZC(K-1)

C *** SURFACE LENGTH OF THE CONTROL VOLUME
         DXN=DXXC(I)
```

```
          DXS=DXXC(I)
          DXF=DXXC(I)
          DXB=DXXC(I)

          DYF=DYYC(J)
          DYB=DYYC(J)
          DYE=DYYC(J)
          DYW=DYYC(J)

          DZE=DZZC(K)
          DZW=DZZC(K)
          DZN=DZZC(K)
          DZS=DZZC(K)

C *** CENTRAL LENGTH OF THE STAGGERED CONTROL VOLUME FOR TEMPERATURE
          DXEE=DXXS(I+2)
          DXE =DXXS(I+1)
          DXW =DXXS(I)
          DXWW=DXXS(I-1)

          DYNN=DYYS(J+2)
          DYN =DYYS(J+1)
          DYS =DYYS(J)
          DYSS=DYYS(J-1)

          DZFF=DZZS(K+2)
          DZF =DZZS(K+1)
          DZB =DZZS(K)
          DZBB=DZZS(K-1)

C *** CALCULATE THE AVERAGE VELOCITY IN THE CENTER OF THE CVOLUME
          UBAR=(U(I+1,J  ,K  )+U(I,J,K))/2.
          VBAR=(V(I  ,J+1,K  )+V(I,J,K))/2.
          WBAR=(W(I  ,J  ,K+1)+W(I,J,K))/2.

C *** CROSS-SECTIONAL AREA OF THE CV AT IT'S CENTER
          DXY=DXI*DYJ
          DYZ=DYJ*DZK
          DZX=DZK*DXI

C *** THE NOMRAL STRESSES
          SIG11(I,J,K)=2.*VIS(I,J,K)*((U(I+1,J,K)-U(I,J,K))/DXI+
     &                 VBAR*(DXN-DXS)/DXY+WBAR*(DXF-DXB)/DZX)
          SIG22(I,J,K)=2.*VIS(I,J,K)*((V(I,J+1,K)-V(I,J,K))/DYJ+
     &                 WBAR*(DYF-DYB)/DYZ+UBAR*(DYE-DYW)/DXY)
          SIG33(I,J,K)=2.*VIS(I,J,K)*((W(I,J,K+1)-W(I,J,K))/DZK+
     &                 UBAR*(DZE-DZW)/DZX+VBAR*(DZN-DZS)/DYZ)
 100  CONTINUE

C *** FOLLOWING DX, DY, DZ, ARE BASED ON THE LOCAL CV FOR SIG12
      DO 200 K=2,NK+4
      DO 200 J=2,NJ+4
      DO 200 I=2,NI+4

C *** CALCULATE THE LENGTH AT VARIOUS POSITIONS
          DXN=DXXS(I)
          DXS=DXXS(I)
          DXI=DXXS(I)

          DXE=DXXC(I)
          DXW=DXXC(I-1)
```

```
      DYE=DYYS(J)
      DYW=DYYS(J)
      DYJ=DYYS(J)

      DYN=DYYC(J)
      DYS=DYYC(J-1)

C *** THE AVERAGE VELOCITY IN THE CONTROL VOLUME
      UBAR=SILIN(U(I,J,K),U(I  ,J-1,K),DYN,DYS)
      VBAR=SILIN(V(I,J,K),V(I-1,J  ,K),DXE,DXW)

C *** AVERAGE VISCOSITY
      VIS12=BILIN(VIS(I  ,J,K),VIS(I  ,J-1,K),DYN,DYS,
     &            VIS(I-1,J,K),VIS(I-1,J-1,K),DYN,DYS,DXE,DXW)

C *** SHEAR STRESS SIG12
      SIGA=((V(I,J,K)-V(I-1,J  ,K))-VBAR*(DYE-DYW)/DYJ)/DXI
      SIGB=((U(I,J,K)-U(I  ,J-1,K))-UBAR*(DXN-DXS)/DXI)/DYJ
      SIG12(I,J,K)=VIS12*(SIGA+SIGB)

C *** FOLLOWING DX, DY, DZ, ARE BASED ON THE LOCAL CV FOR SIG13

C *** CALCULATE THE LENGTH AT VARIOUS POSITIONS
      DXF=DXXS(I)
      DXB=DXXS(I)
      DXI=DXXS(I)

      DXE=DXXC(I)
      DXW=DXXC(I-1)

      DZE=DZZS(K)
      DZW=DZZS(K)
      DZK=DZZS(K)

      DZF=DZZC(K)
      DZB=DZZC(K-1)

C *** THE AVERAGE VELOCITY IN THE CONTROL VOLUME
      UBAR=SILIN(U(I,J,K),U(I  ,J,K-1),DZF,DZB)
      WBAR=SILIN(W(I,J,K),W(I-1,J,K  ),DXE,DXW)

C *** AVERAGE VISCOSITY
      VIS13=BILIN(VIS(I  ,J,K),VIS(I  ,J,K-1),DZF,DZB,
     &            VIS(I-1,J,K),VIS(I-1,J,K-1),DZF,DZB,DXE,DXW)

C *** SHEAR STRESS SIG13
      SIGA=((W(I,J,K)-W(I-1,J,K  ))-WBAR*(DZE-DZW)/DZK)/DXI
      SIGB=((U(I,J,K)-U(I  ,J,K-1))-UBAR*(DXF-DXB)/DXI)/DZK
      SIG13(I,J,K)=VIS13*(SIGA+SIGB)

C *** FOLLOWING DX, DY, DZ, ARE BASED ON THE LOCAL CV FOR SIG23

C *** LENGTH AT VARIOUS POSITIONS
      DYF=DYYS(J)
      DYB=DYYS(J)
      DYJ=DYYS(J)

      DYN=DYYC(J)
      DYS=DYYC(J-1)
```

```
          DZN=DZZS(K)
          DZS=DZZS(K)
          DZK=DZZS(K)

          DZF=DZZC(K)
          DZB=DZZC(K-1)

C *** THE AVERAGE VELOCITY IN THE CONTROL VOLUME
          WBAR=SILIN(W(I,J,K),W(I,J-1,K  ),DYN,DYS)
          VBAR=SILIN(V(I,J,K),V(I,J  ,K-1),DZF,DZB)

C *** AVERAGE VISCOSITY
          VIS23=BILIN(VIS(I,J,K  ),VIS(I,J-1,K  ),DYN,DYS,
     &                VIS(I,J,K-1),VIS(I,J-1,K-1),DYN,DYS,DZF,DZB)

          SIGA=((V(I,J,K)-V(I,J  ,K-1))-VBAR*(DYF-DYB)/DYJ)/DZK
          SIGB=((W(I,J,K)-W(I,J-1,K  ))-WBAR*(DZN-DZS)/DZK)/DYJ
          SIG23(I,J,K)=VIS23*(SIGA+SIGB)

  200 CONTINUE
      RETURN
      END

************************************************************************
************************************************************************
      SUBROUTINE TCP
************************************************************************
*THIS SUBROUTINE CALCULATES THE NONDIMENSIONAL TEMPERATURE AT THE
*THERMOCOUPLE POSITIONS.
************************************************************************

      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      COMMON/R4/XC(40),YC(40),ZC(40),XS(40),YS(40),ZS(40),DXXC(40),
     &          DYYC(40),DZZC(40),DXXS(40),DYYS(40),DZZS(40)
      COMMON/BL1/DX,DY,DZ,DTIME,TCOOL,PI,Q,QR
      COMMON/BL32/T(30,30,25),R(30,30,25),P(30,30,25),C(30,30,25),
     &            U(30,30,25),V(30,30,25),W(30,30,25)
      COMMON/BL38/TCOUP(30),CX(30),CY(30),CZ(30),NTH(30,3),NTHCO

C *** CALCULATE SIZE OF CONTROL VOLUME CONTAINING THE THERMOCOUPLES
      DO 5100 N=1,NTHCO
         VOL=ABS((XC(NTH(N,1)+1)-XC(NTH(N,1)))*(YC(NTH(N,2)+1)-
     &       YC(NTH(N,2)))*(ZC(NTH(N,3)+1)-ZC(NTH(N,3))))
         TCOUP(N)=0.

         DO 5101 I=NTH(N,1),NTH(N,1)+1
            II=2*NTH(N,1)+1-I
            DO 5102 J=NTH(N,2),NTH(N,2)+1
               JJ=2*NTH(N,2)+1-J
               DO 5103 K=NTH(N,3),NTH(N,3)+1
                  KK=2*NTH(N,3)+1-K

C *** CORRECT TEMPERATURES FOR THERMOCOUPLES NOT LOCATED ON NODES
                  TVOL=ABS((XC(I)-CX(N))*(YC(J)-CY(N))*(ZC(K)-CZ(N)))
                  WVOL=TVOL/VOL
                  TCOUP(N)=TCOUP(N)+WVOL*T(II,JJ,KK)
 5103          CONTINUE
 5102       CONTINUE
 5101    CONTINUE
         IF (TCOUP(N).LT.TCOOL) TCOUP(N)=TCOOL
 5100 CONTINUE
```

```
      RETURN
      END

************************************************************************
************************************************************************
      SUBROUTINE TRID(IST,JST,KST,ISP,JSP,KSP,PHI)
************************************************************************

      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      COMMON/BL7/NI,NJ,NK,KRUN,NBLOR,NWRP
      DIMENSION A(99),B(99),C(99),PHI(30,30,25)
      COMMON/BL36/AP(30,30,25),AE(30,30,25),AW(30,30,25),AN(30,30,25),
     &            AS(30,30,25),AF(30,30,25),AB(30,30,25),SP(30,30,25),
     &            SU(30,30,25),RI(30,30,25)

C *** FORWARD SWEEP IN THE X DIRECTION (FROM IST TO ISP)
      A(IST-1)=0.
      C(IST-1)=0.
      DO 100 J=JST,JSP
      DO 100 K=KST,KSP
         DO 101 I=IST,ISP
            A(I)=AE(I,J,K)
            B(I)=AW(I,J,K)
            C(I)=AN(I,J,K)*PHI(I,J+1,K)+AS(I,J,K)*PHI(I,J-1,K)+
     &           AF(I,J,K)*PHI(I,J,K+1)+AB(I,J,K)*PHI(I,J,K-1)+SU(I,J,K)
            TERM=1./(AP(I,J,K)-B(I)*A(I-1))
            IF (ABS(A(I)).LE.1.0E-38) A(I)=0.0
            IF (ABS(B(I)).LE.1.0E-38) B(I)=0.0
            IF (ABS(C(I)).LE.1.0E-38) C(I)=0.0
            IF (ABS(TERM).LE.1.0E-38) TERM=0.0
            A(I)=A(I)*TERM
            C(I)=(C(I)+B(I)*C(I-1))*TERM
  101    CONTINUE
         PHI(ISP,J,K)=C(ISP)
         DO 102 I=ISP-1,IST,-1
            PHI(I,J,K)=A(I)*PHI(I+1,J,K)+C(I)
  102    CONTINUE
  100 CONTINUE

C *** FORWARD SWEEP IN THE Y DIRECTION (FROM JST TO JSP)
      A(JST-1)=0.
      C(JST-1)=0.
      DO 200 K=KST,KSP
      DO 200 I=IST,ISP
         DO 201 J=JST,JSP
            A(J)=AN(I,J,K)
            B(J)=AS(I,J,K)
            C(J)=AE(I,J,K)*PHI(I+1,J,K)+AW(I,J,K)*PHI(I-1,J,K)+
     &           AF(I,J,K)*PHI(I,J,K+1)+AB(I,J,K)*PHI(I,J,K-1)+SU(I,J,K)
            TERM=1./(AP(I,J,K)-B(J)*A(J-1))
            IF (ABS(A(J)).LE.1.0E-38) A(J)=0.0
            IF (ABS(B(J)).LE.1.0E-38) B(J)=0.0
            IF (ABS(C(J)).LE.1.0E-38) C(J)=0.0
            IF (ABS(TERM).LE.1.0E-38) TERM=0.0
            A(J)=A(J)*TERM
            C(J)=(C(J)+B(J)*C(J-1))*TERM
  201    CONTINUE
         PHI(I,JSP,K)=C(JSP)
         DO 202 J=JSP-1,JST,-1
            PHI(I,J,K)=A(J)*PHI(I,J+1,K)+C(J)
```

```
 202    CONTINUE
 200 CONTINUE

C *** FORWARD SWEEP IN THE Z DIRECTION (FROM KST TO KSP)
     A(KST-1)=0.
     C(KST-1)=0.
     DO 300 I=IST,ISP
     DO 300 J=JST,JSP
        DO 301 K=KST,KSP
           A(K)=AF(I,J,K)
           B(K)=AB(I,J,K)
           C(K)=AE(I,J,K)*PHI(I+1,J,K)+AW(I,J,K)*PHI(I-1,J,K)+
    &           AN(I,J,K)*PHI(I,J+1,K)+AS(I,J,K)*PHI(I,J-1,K)+SU(I,J,K)
           TERM=1./(AP(I,J,K)-B(K)*A(K-1))
           IF (ABS(A(K)).LE.1.0E-38) A(K)=0.0
           IF (ABS(B(K)).LE.1.0E-38) B(K)=0.0
           IF (ABS(C(K)).LE.1.0E-38) C(K)=0.0
           IF (ABS(TERM).LE.1.0E-38) TERM=0.0
           A(K)=A(K)*TERM
           C(K)=(C(K)+B(K)*C(K-1))*TERM
 301    CONTINUE
        PHI(I,J,KSP)=C(KSP)
        DO 302 K=KSP-1,KST,-1
           PHI(I,J,K)=A(K)*PHI(I,J,K+1)+C(K)
 302    CONTINUE
 300 CONTINUE

C *** REVERSE SWEEP IN X DIRECTION (FROM ISP TO IST)
     B(KSP+1)=0.
     C(KSP+1)=0.
     DO 600 I=ISP,IST,-1
     DO 600 J=JSP,JST,-1
        DO 601 K=KSP,KST,-1
           A(K)=AF(I,J,K)
           B(K)=AB(I,J,K)
           C(K)=AE(I,J,K)*PHI(I+1,J,K)+AW(I,J,K)*PHI(I-1,J,K)+
    &           AN(I,J,K)*PHI(I,J+1,K)+AS(I,J,K)*PHI(I,J-1,K)+SU(I,J,K)
           TERM=1./(AP(I,J,K)-A(K)*B(K+1))
           B(K)=B(K)*TERM
           C(K)=(C(K)+A(K)*C(K+1))*TERM
           IF (ABS(A(K)).LE.1.0E-38) A(K)=0.0
           IF (ABS(B(K)).LE.1.0E-38) B(K)=0.0
           IF (ABS(C(K)).LE.1.0E-38) C(K)=0.0
 601    CONTINUE
        PHI(I,J,KST)=C(KST)
        DO 602 K=KST+1,KSP
           PHI(I,J,K)=B(K)*PHI(I,J,K-1)+C(K)
 602    CONTINUE
 600 CONTINUE

C *** REVERSE SWEEP IN THE Y DIRECTION (FROM JSP TO JST)
     B(JSP+1)=0.
     C(JSP+1)=0.
     DO 500 K=KSP,KST,-1
     DO 500 I=ISP,IST,-1
        DO 501 J=JSP,JST,-1
           A(J)=AN(I,J,K)
           B(J)=AS(I,J,K)
           C(J)=AE(I,J,K)*PHI(I+1,J,K)+AW(I,J,K)*PHI(I-1,J,K)+
    &           AF(I,J,K)*PHI(I,J,K+1)+AB(I,J,K)*PHI(I,J,K-1)+SU(I,J,K)
           TERM=1./(AP(I,J,K)-A(J)*B(J+1))
```

```
            B(J)=B(J)*TERM
            C(J)=(C(J)+A(J)*C(J+1))*TERM
            IF (ABS(A(J)).LE.1.0E-38) A(J)=0.0
            IF (ABS(B(J)).LE.1.0E-38) B(J)=0.0
            IF (ABS(C(J)).LE.1.0E-38) C(J)=0.0
  501    CONTINUE
         PHI(I,JST,K)=C(JST)
         DO 502 J=JST+1,JSP
            PHI(I,J,K)=B(J)*PHI(I,J-1,K)+C(J)
  502    CONTINUE
  500 CONTINUE

C *** REVERSE SWEEP IN THE Z DIRECTION (FROM KSP TO KST)
      B(ISP+1)=0.
      C(ISP+1)=0.
      DO 400 J=JSP,JST,-1
      DO 400 K=KSP,KST,-1
         DO 401 I=ISP,IST,-1
            A(I)=AE(I,J,K)
            B(I)=AW(I,J,K)
            C(I)=AN(I,J,K)*PHI(I,J+1,K)+AS(I,J,K)*PHI(I,J-1,K)+
     &           AF(I,J,K)*PHI(I,J,K+1)+AB(I,J,K)*PHI(I,J,K-1)+SU(I,J,K)
            TERM=1./(AP(I,J,K)-A(I)*B(I+1))
            B(I)=B(I)*TERM
            C(I)=(C(I)+A(I)*C(I+1))*TERM
            IF (ABS(A(I)).LE.1.0E-38) A(I)=0.0
            IF (ABS(B(I)).LE.1.0E-38) B(I)=0.0
            IF (ABS(C(I)).LE.1.0E-38) C(I)=0.0
  401    CONTINUE
         PHI(IST,J,K)=C(IST)
         DO 402 I=IST+1,ISP
            PHI(I,J,K)=B(I)*PHI(I-1,J,K)+C(I)
  402    CONTINUE
  400 CONTINUE

      RETURN
      END

************************************************************************
************************************************************************
      SUBROUTINE VIEW
************************************************************************
*  NTHS          = TOTAL NUMBER OF NODES CONTAINING HEAT SORCE
*  NFX,NFY,NFZ = NUMBER OF NODES IN HEAT SOURCE PER DIRECTION
*  FX,FY,FZ      = STARTING COORDINATES OF HEAT SOURCE
*  DXS,DYS,DZS = LENGTH IN EACH DIRECTION ON THE SOUTH SURFACE
*                OF THE ENCLOSURE
*  SXS,SYS,SZS = COORDINATES OF THE SURFACE ELEMENT (USED TO
*                CALCULATE THE DISTANCE TO HEAT SOURCE)
*  VFHSW,VFHSE = VIEW FACTOR FROM THE HEAT SOURCE TO WEST, EAST
*                SURFACES OF THE ENCLOSURE
*  RSQW,RSQE    = SQUARE OF DISTANCE FROM THE HEAT SOURCE TO THE
*                WEST, EAST SURFACE ELEMENT.
*
*                NODES 2 THRU NI,2 THRU NJ AND 2 THRU NK ARE CONTAINED
*                INSIDE THE WALL, RADIATION EFFECTS INVOLVE ONLY NODES
*                3 THRU NIM1, 3 THRU NJM1, AND 3 THRU NKM1
*
* VFHSBW(N,M,I,J),
* VFHSBE(N,M,I,J)  = VIEW FACTOR FROM FIRE NODE N TO ELEMENT (I,J) OF
*                    WEST, EAST SURFACES OF CV M.
```

```
*********************************************************************
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      COMMON/R4/XC(40),YC(40),ZC(40),XS(40),YS(40),ZS(40),DXXC(40),
     &          DYYC(40),DZZC(40),DXXS(40),DYYS(40),DZZS(40)
      COMMON/BL1/DX,DY,DZ,DTIME,TCOOL,PI,Q,QR
      COMMON/BL2/X,Y,H,TFLR,TWAL
      COMMON/BL7/NI,NJ,NK,KRUN,NBLOR,NWRP
      COMMON/BL22/CPS(20),CONS(20),WFAN(20),NCHIP,ICHPB(20),NCHPI(20),
     &            JCHPB(20),NCHPJ(20),KCHPB(20),NCHPK(20)
      COMMON/BL37/VIS(30,30,25),COND(30,30,25),RESORM(40),
     &            CPM(30,30,25),NHSZ(3,2),NOD(30,30,25)
      COMMON/BL40/VFHSW(8,30,30),VFHSE(8,30,30),VFHSS(8,30,30),
     &            VFHSN(8,30,30),VFHSB(8,30,30),VFHSF(8,30,30)
      COMMON/BL41/VFHSBW(8,8,34,34),VFHSBE(8,8,34,34),VFHSBS(8,8,34,34),
     &            VFHSBN(8,8,34,34),VFHSBB(8,8,34,34),VFHSBF(8,8,34,34)

      NTHS=NHSZ(3,2)-NHSZ(3,1)+1
      DO 500 N=1,NTHS
      SUMM=0.
      NFX=NHSZ(1,1)
      NFY=NHSZ(2,1)
      NFZ=NHSZ(3,1)-1+N
      FX =XS(NFX+1)
      FY =YS(NFY+1)
      FZ =ZS(NFZ+1)

C *** VIEW FACTOR FROM FIRE TO WEST & EAST SURFACES OF ENCLOSURE
      DO 100 J=3,NJ+2
      DO 100 K=3,NK+2
         DYW=DYYC(J)
         DYE=DYYC(J)
         DZW=DZZC(K)
         DZE=DZZC(K)

         SXW=XS(3)
         SXE=XS(NI+3)
         SYW=YC(J)
         SYE=YC(J)
         SZW=ZC(K)
         SZE=ZC(K)

         DYZW=DYW*DZW
         DYZE=DYE*DZE

         RSQW=(FX-SXW)**2+(FY-SYW)**2+(FZ-SZW)**2
         RSQE=(FX-SXE)**2+(FY-SYE)**2+(FZ-SZE)**2

         VFHSW(N,J,K)=SQRT((FX-SXW)**2/RSQW)*DYZW/(4.0*PI*RSQW)
         VFHSE(N,J,K)=SQRT((FX-SXE)**2/RSQE)*DYZE/(4.0*PI*RSQE)
         SUMM=SUMM+VFHSW(N,J,K)+VFHSE(N,J,K)
  100 CONTINUE

C *** VIEW FACTOR FROM FIRE TO NORTH & SOUTH SURFACES OF ENCLOSURE
      DO 200 I=3,NI+2
      DO 200 K=3,NK+2
         DXS=DXXC(I)
         DXN=DXXC(I)
         DZS=DZZC(K)
         DZN=DZZC(K)
```

```
         SXN=XC(I)
         SXS=XC(I)
         SYN=YS(NJ+3)
         SYS=YS(3)
         SZN=ZC(K)
         SZS=ZC(K)

         DZXS=DXS*DZS
         DZXN=DXN*DZN

         RSQS=(FX-SXS)**2+(FY-SYS)**2+(FZ-SZS)**2
         RSQN=(FX-SXN)**2+(FY-SYN)**2+(FZ-SZN)**2

         VFHSS(N,K,I)=SQRT((FY-SYS)**2/RSQS)*DZXS/(4.0*PI*RSQS)
         VFHSN(N,K,I)=SQRT((FY-SYN)**2/RSQN)*DZXN/(4.0*PI*RSQN)
         SUMM=SUMM+VFHSS(N,K,I)+VFHSN(N,K,I)
  200 CONTINUE

C *** VIEW FACTOR FROM FIRE TO FRONT & BACK SURFACES OF ENCLOSURE
      DO 300 I=3,NI+2
      DO 300 J=3,NJ+2
         DXF=DXXC(I)
         DXB=DXXC(I)
         DYF=DYYC(J)
         DYB=DYYC(J)

         SXF=XC(I)
         SXB=XC(I)
         SYF=YC(J)
         SYB=YC(J)
         SZF=ZS(NK+3)
         SZB=ZS(3)

         DXYB=DXB*DYB
         DXYF=DXF*DYF

         RSQB=(FX-SXB)**2+(FY-SYB)**2+(FZ-SZB)**2
         RSQF=(FX-SXF)**2+(FY-SYF)**2+(FZ-SZF)**2

         VFHSB(N,I,J)=SQRT((FZ-SZB)**2/RSQB)*DXYB/(4.0*PI*RSQB)
         VFHSF(N,I,J)=SQRT((FZ-SZF)**2/RSQF)*DXYF/(4.0*PI*RSQF)
         SUMM=SUMM+VFHSB(N,I,J)+VFHSF(N,I,J)
  300 CONTINUE

C *** MODIFY VIEW FACTORS SO THEIR SUMMATION EQUALS UNITY.
      DO 150 J=3,NJ+2
      DO 150 K=3,NK+2
         VFHSW(N,J,K)=VFHSW(N,J,K)/SUMM
         VFHSE(N,J,K)=VFHSE(N,J,K)/SUMM
  150 CONTINUE
      DO 250 K=3,NK+2
      DO 250 I=3,NI+2
         VFHSS(N,K,I)=VFHSS(N,K,I)/SUMM
         VFHSN(N,K,I)=VFHSN(N,K,I)/SUMM
  250 CONTINUE
      DO 350 I=3,NI+2
      DO 350 J=3,NJ+2
         VFHSB(N,I,J)=VFHSB(N,I,J)/SUMM
         VFHSF(N,I,J)=VFHSF(N,I,J)/SUMM
  350 CONTINUE
```

```
      IF (NCHIP.LT.NBLOR) GOTO 500

C *** CALCULATE VIEW FACTORS FROM FIRE N TO INTERNAL SOLID BLOCKS
C *** (BLOCKS ARE ONLY THOSE NOT INCLUDED IN THE WALL)

C *** VIEW FACTOR FROM THE FIRE TO WEST & EAST SURFACES OF BLOCK M
      DO 900 M=1,NCHIP-NBLOR+1
         IB  =ICHPB(M+NBLOR-1)
         IE  =IB+NCHPI(M+NBLOR-1)-1
         JB  =JCHPB(M+NBLOR-1)
         JE  =JB+NCHPJ(M+NBLOR-1)-1
         KB  =KCHPB(M+NBLOR-1)
         KE  =KB+NCHPK(M+NBLOR-1)-1

         SXW=XS(IB)
         SXE=XS(IE)

         DO 400 JJ=JB,JE-1
         DO 400 KK=KB,KE-1
            DYW=DYYC(JJ)
            DYE=DYYC(JJ)
            DZW=DZZC(KK)
            DZE=DZZC(KK)

            SYW=YC(JJ)
            SYE=YC(JJ)
            SZW=ZC(KK)
            SZE=ZC(KK)

            DYZW=DYW*DZW
            DYZE=DYE*DZE

            RSQW=(FX-SXW)**2+(FY-SYW)**2+(FZ-SZW)**2
            RSQE=(FX-SXE)**2+(FY-SYE)**2+(FZ-SZE)**2
            VFHSBW(N,M,JJ,KK)=SQRT((FX-SXW)**2/RSQW)*DYZW/(4.0*PI*RSQW)
            VFHSBE(N,M,JJ,KK)=SQRT((FX-SXE)**2/RSQE)*DYZE/(4.0*PI*RSQE)

C *** MODIFY VIEW FACTORS DUE TO INTRODUCTION OF INTERNAL BLOCKS
            IF (SXE.LT.FX) VFHSBW(N,M,JJ,KK)=0.
            IF (SXW.GT.FX) VFHSBE(N,M,JJ,KK)=0.

C *** THE FIRE CAN'T SEE THE WEST AND EAST SURFACES OF THE BLOCK
            IF (SXW.LE.FX.AND.SXE.GE.FX) THEN
               VFHSBW(N,M,JJ,KK)=0.
               VFHSBE(N,M,JJ,KK)=0.
            ENDIF
  400    CONTINUE

C *** CHECK TO SEE IF ANY ELEMENT ON THE WALL IS SHADED BY A SOLID
BLOCK.

C *** CHECK WEST AND EAST WALLS OF ENCLOSURE
         DO 410 J=3,NJ+2
         DO 410 K=3,NK+2

C *** THE BLOCK IS ON THE WEST SIDE OF THE FIRE
            IF (SXE.LT.FX) THEN
               NVIW=NVIWX(FX,FY,FZ,XS(3),YC(J),ZC(K),SXE,
     &                    IB,JB,KB,IE,JE,KE)
               IF (NVIW.EQ.1) VFHSW(N,J,K)=0.
            ENDIF
```

```
C *** THE BLOCK IS ON THE EAST SIDE OF THE FIRE
            IF (SXW.GT.FX) THEN
               NVIW=NVIWX(FX,FY,FZ,XS(NI+3),YC(J),ZC(K),SXW,
     &                   IB,JB,KB,IE,JE,KE)
               IF (NVIW.EQ.1) VFHSE(N,J,K)=0.
            ENDIF
  410    CONTINUE

C *** CHECK SOUTH AND NORTH WALLS OF THE ENCLOSURE
         DO 420 K=3,NK+2
         DO 420 I=3,NI+2

C *** THE BLOCK IS ON THE WEST SIDE OF THE FIRE
            IF (SXE.LT.FX.AND.XC(I).LT.FX) THEN
               NVIW=NVIWX(FX,FY,FZ,XC(I),YS(3),ZC(K),SXE,
     &                   IB,JB,KB,IE,JE,KE)
               IF (NVIW.EQ.1) VFHSS(N,K,I)=0.
               NVIW=NVIWX(FX,FY,FZ,XC(I),YS(NJ+3),ZC(K),SXE,
     &                   IB,JB,KB,IE,JE,KE)
               IF (NVIW.EQ.1) VFHSN(N,K,I)=0.
            ENDIF

C *** THE BLOCK IS ON THE EAST SIDE OF THE FIRE
            IF (SXW.GT.FX.AND.XC(I).GT.FX) THEN
               NVIW=NVIWX(FX,FY,FZ,XC(I),YS(3),ZC(K),SXW,
     &                   IB,JB,KB,IE,JE,KE)
               IF (NVIW.EQ.1) VFHSS(N,K,I)=0.
               NVIW=NVIWX(FX,FY,FZ,XC(I),YS(NJ+3),ZC(K),SXW,
     &                   IB,JB,KB,IE,JE,KE)
               IF (NVIW.EQ.1) VFHSN(N,K,I)=0.
            ENDIF
  420    CONTINUE

C *** CHECK BACK AND FRONT WALLS OF ENCLOSURE
         DO 430 I=3,NI+2
         DO 430 J=3,NJ+2

C *** THE BLOCK IS ON THE WEST SIDE OF THE FIRE
            IF (SXE.LT.FX.AND.XC(I).LT.FX) THEN
               NVIW=NVIWX(FX,FY,FZ,XC(I),YC(J),ZS(3),SXE,
     &                   IB,JB,KB,IE,JE,KE)
               IF (NVIW.EQ.1) VFHSB(N,I,J)=0.
               NVIW=NVIWX(FX,FY,FZ,XC(I),YC(J),ZS(NK+3),SXE,
     &                   IB,JB,KB,IE,JE,KE)
               IF (NVIW.EQ.1) VFHSF(N,I,J)=0.
            ENDIF

C *** THE BLOCK IS ON THE EAST SIDE OF THE FIRE
            IF (SXW.GT.FX.AND.XC(I).GT.FX) THEN
               NVIW=NVIWX(FX,FY,FZ,XC(I),YC(J),ZS(3),SXW,
     &                   IB,JB,KB,IE,JE,KE)
               IF (NVIW.EQ.1) VFHSB(N,I,J)=0.
               NVIW=NVIWX(FX,FY,FZ,XC(I),YC(J),ZS(NK+3),SXW,
     &                   IB,JB,KB,IE,JE,KE)
               IF (NVIW.EQ.1) VFHSF(N,I,J)=0.
            ENDIF
  430    CONTINUE

C *** VIEW FACTOR FROM FIRE TO NORTH & SOUTH SURFACES OF BLOCK M
         DO 600 II=IB,IE-1
```

```
        DO 600 KK=KB,KE-1
           DXN=DXXC(II)
           DXS=DXXC(II)
           DZN=DZZC(KK)
           DZS=DZZC(KK)

           SXN=XC(II)
           SXS=XC(II)
           SYN=YS(JE)
           SYS=YS(JB)
           SZN=ZC(KK)
           SZS=ZC(KK)

           DZXN=DXN*DZN
           DZXS=DXS*DZS

           RSQS=(FX-SXS)**2+(FY-SYS)**2+(FZ-SZS)**2
           RSQN=(FX-SXN)**2+(FY-SYN)**2+(FZ-SZN)**2
           VFHSBS(N,M,KK,II)=SQRT((FY-SYS)**2/RSQS)*DZXS/(4.0*PI*RSQS)
           VFHSBN(N,M,KK,II)=SQRT((FY-SYN)**2/RSQN)*DZXN/(4.0*PI*RSQN)

C *** MODIFY VIEW FACTORS DUE TO INTRODUCTION OF INTERNAL BLOCKS
           IF (SYN.LT.FY) VFHSBS(N,M,KK,II)=0.
           IF (SYS.GT.FY) VFHSBN(N,M,KK,II)=0.
           IF (SYS.LE.FY.AND.SYN.GE.FY) THEN
              VFHSBS(N,M,KK,II)=0.
              VFHSBN(N,M,KK,II)=0.
           ENDIF
  600   CONTINUE

C *** CHECK IF ANY ELEMENT ON WALL IS SHADED BY INTERNAL SOLID BLOCKS

C *** CHECK WEST AND EAST WALLS OF THE ENCLOSURE
        DO 610 J=3,NJ+2
        DO 610 K=3,NK+2

C *** THE BLOCK IS ON THE SOUTH SIDE OF THE FIRE
           IF (SYN.LT.FY.AND.YC(J).LT.FY) THEN
              NVIW=NVIWY(FX,FY,FZ,XS(3),YC(J),ZC(K),SYN,
     &                   IB,JB,KB,IE,JE,KE)
              IF (NVIW.EQ.1) VFHSW(N,J,K)=0.
              NVIW=NVIWY(FX,FY,FZ,XS(NI+3),YC(J),ZC(K),SYN,
     &                   IB,JB,KB,IE,JE,KE)
              IF (NVIW.EQ.1) VFHSE(N,J,K)=0.
           ENDIF

C *** THE BLOCK IS ON THE NORTH SIDE OF THE FIRE
           IF (SYS.GT.FY.AND.YC(J).GT.FY) THEN
              NVIW=NVIWY(FX,FY,FZ,XS(3),YC(J),ZC(K),SYS,
     &                   IB,JB,KB,IE,JE,KE)
              IF (NVIW.EQ.1) VFHSW(N,J,K)=0.
              NVIW=NVIWY(FX,FY,FZ,XS(NI+3),YC(J),ZC(K),SYS,
     &                   IB,JB,KB,IE,JE,KE)
              IF (NVIW.EQ.1) VFHSE(N,J,K)=0.
           ENDIF
  610   CONTINUE

C *** CHECK THE SOUTH AND NORTH WALLS OF THE ENCLOSURE
        DO 620 K=3,NK+2
        DO 620 I=3,NI+2
```

```
C *** THE BLOCK IS ON THE SOUTH SIDE OF THE FIRE
            IF (SYN.LT.FY) THEN
               NVIW=NVIWY(FX,FY,FZ,XC(I),YS(3),ZC(K),SYN,
     &                    IB,JB,KB,IE,JE,KE)
               IF (NVIW.EQ.1) VFHSS(N,K,I)=0.
            ENDIF

C *** THE BLOCK IS ON THE NORTH SIDE OF THE FIRE
            IF (SYS.GT.FY) THEN
               NVIW=NVIWY(FX,FY,FZ,XC(I),YS(NJ+3),ZC(K),SYS,
     &                    IB,JB,KB,IE,JE,KE)
               IF (NVIW.EQ.1) VFHSN(N,K,I)=0.
            ENDIF
  620    CONTINUE

C *** THE BACK AND FRONT WALLS OF THE ENCLOSURE
         DO 630 I=3,NI+2
         DO 630 J=3,NJ+2

C *** THE BLOCK IS ON THE SOUTH SIDE OF THE FIRE
            IF (SYN.LT.FY.AND.YC(J).LT.FY) THEN
               NVIW=NVIWY(FX,FY,FZ,XC(I),YC(J),ZS(3),SYN,
     &                    IB,JB,KB,IE,JE,KE)
               IF (NVIW.EQ.1) VFHSB(N,I,J)=0.
               NVIW=NVIWY(FX,FY,FZ,XC(I),YC(J),ZS(NK+3),SYN,
     &                    IB,JB,KB,IE,JE,KE)
               IF (NVIW.EQ.1) VFHSF(N,I,J)=0.
            ENDIF

C *** THE BLOCK IS ON THE NORTH SIDE OF THE FIRE
            IF (SYS.GT.FY.AND.YC(J).GT.FY) THEN
               NVIW=NVIWY(FX,FY,FZ,XC(I),YC(J),ZS(3),SYS,
     &                    IB,JB,KB,IE,JE,KE)
               IF (NVIW.EQ.1) VFHSB(N,I,J)=0.
               NVIW=NVIWY(FX,FY,FZ,XC(I),YC(J),ZS(NK+3),SYS,
     &                    IB,JB,KB,IE,JE,KE)
               IF (NVIW.EQ.1) VFHSF(N,I,J)=0.
            ENDIF
  630    CONTINUE

C *** CHECK VIEW FACTORS FROM FIRE TO BACK & FRONT SURFACES OF BLOCK M
         DO 700 II=IB,IE-1
         DO 700 JJ=JB,JE-1
            DXF=DXXC(II)
            DXB=DXXC(II)
            DYF=DYYC(JJ)
            DYB=DYYC(JJ)

            DXYB=DXB*DYB
            DXYF=DXF*DYF

            SXF=XC(II)
            SXB=XC(II)
            SYF=YC(JJ)
            SYB=YC(JJ)
            SZF=ZS(KE)
            SZB=ZS(KB)

            RSQB=(FX-SXB)**2+(FY-SYB)**2+(FZ-SZB)**2
            RSQF=(FX-SXF)**2+(FY-SYF)**2+(FZ-SZF)**2
            VFHSBB(N,M,II,JJ)=(FZ-SZB)**2*DXYB/(4.0*PI*RSQB**2)
```

```
            VFHSBF(N,M,II,JJ)=(FZ-SZF)**2*DXYF/(4.0*PI*RSQF**2)

C *** MODIFY VIEW FACTORS DUE TO INTRODUCTION OF INTERNAL SOLID BLOCKS
            IF (SZF.LT.FZ) VFHSBB(N,M,II,JJ)=0.
            IF (SZB.GT.FZ) VFHSBF(N,M,II,JJ)=0.
            IF (SZB.LE.FZ.AND.SZF.GE.FZ) THEN
               VFHSBB(N,M,II,JJ)=0.
               VFHSBF(N,M,II,JJ)=0.
            ENDIF
  700    CONTINUE

C *** CHECK IF ANY ELEMENT ON THE WALL IS SHADED BY SOLID BLOCK.

C *** THE WEST AND EAST WALLS OF THE ENCLOSURE
         DO 710 J=3,NJ+2
         DO 710 K=3,NK+2

C *** THE BLOCK IS ON THE BACK SIDE OF THE FIRE
            IF (SZF.LT.FZ.AND.ZC(K).LT.FZ) THEN
               NVIW=NVIWZ(FX,FY,FZ,XS(3),YC(J),ZC(K),SZF,
     &                    IB,JB,KB,IE,JE,KE)
               IF (NVIW.EQ.1) VFHSW(N,J,K)=0.
               NVIW=NVIWZ(FX,FY,FZ,XS(NI+3),YC(J),ZC(K),SZF,
     &                    IB,JB,KB,IE,JE,KE)
               IF (NVIW.EQ.1) VFHSE(N,J,K)=0.
            ENDIF

C *** THE BLOCK IS ON THE FRONT SIDE OF THE FIRE
            IF (SZB.GT.FZ.AND.ZC(K).GT.FZ) THEN
               NVIW=NVIWZ(FX,FY,FZ,XS(3),YC(J),ZC(K),SZB,
     &                    IB,JB,KB,IE,JE,KE)
               IF (NVIW.EQ.1) VFHSW(N,J,K)=0.
               NVIW=NVIWZ(FX,FY,FZ,XS(NI+3),YC(J),ZC(K),SZB,
     &                    IB,JB,KB,IE,JE,KE)
               IF (NVIW.EQ.1) VFHSE(N,J,K)=0.
            ENDIF
  710    CONTINUE

C *** CHECK THE SOUTH AND NORTH WALLS OF THE ENCLOSURE
         DO 720 K=3,NK+2
         DO 720 I=3,NI+2

C *** THE BLOCK IS ON THE BACK SIDE OF THE FIRE
            IF (SZF.LT.FZ.AND.ZC(K).LT.FZ) THEN
               NVIW=NVIWZ(FX,FY,FZ,XC(I),YS(3),ZC(K),SZF,
     &                    IB,JB,KB,IE,JE,KE)
               IF (NVIW.EQ.1) VFHSS(N,K,I)=0.
               NVIW=NVIWZ(FX,FY,FZ,XC(I),YS(NJ+3),ZC(K),SZF,
     &                    IB,JB,KB,IE,JE,KE)
               IF (NVIW.EQ.1) VFHSN(N,K,I)=0.
            ENDIF

C *** THE BLOCK IS ON THE FRONT SIDE OF THE FIRE
            IF (SZB.GT.FZ.AND.ZC(K).GT.FZ) THEN
               NVIW=NVIWZ(FX,FY,FZ,XC(I),YS(3),ZC(K),SZB,
     &                    IB,JB,KB,IE,JE,KE)
               IF (NVIW.EQ.1) VFHSS(N,K,I)=0.
               NVIW=NVIWZ(FX,FY,FZ,XC(I),YS(NJ+3),ZC(K),SZB,
     &                    IB,JB,KB,IE,JE,KE)
               IF (NVIW.EQ.1) VFHSN(N,K,I)=0.
            ENDIF
```

```
  720    CONTINUE

C *** CHECK THE BACK AND FRONT WALLS OF THE ENCLOSURE
         DO 730 I=3,NI+2
         DO 730 J=3,NJ+2

C *** THE BLOCK IS ON THE BACK SIDE OF THE FIRE
            IF (SZF.LT.FZ) THEN
               NVIW=NVIWZ(FX,FY,FZ,XC(I),YC(J),ZS(3),SZF,
     &                    IB,JB,KB,IE,JE,KE)
               IF (NVIW.EQ.1) VFHSB(N,I,J)=0.
            ENDIF

C *** THE BLOCK IS ON THE FRONT SIDE OF THE FIRE
            IF (SZB.GT.FZ) THEN
               NVIW=NVIWZ(FX,FY,FZ,XC(I),YC(J),ZS(NK+3),SZB,
     &                    IB,JB,KB,IE,JE,KE)
               IF (NVIW.EQ.1) VFHSF(N,I,J)=0.
            ENDIF
  730    CONTINUE
  900 CONTINUE
  500 CONTINUE

      RETURN
      END

***********************************************************************
***********************************************************************
      FUNCTION BILIN(V1,V2,D1,D2,V3,V4,D3,D4,D5,D6)
***********************************************************************
* BI-LINEAR INTERPOLATION

      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      V12=(V1*D2+V2*D1)/(D1+D2)
      V34=(V3*D4+V4*D3)/(D3+D4)
      BILIN=(V12*D6+V34*D5)/(D5+D6)

      RETURN
      END

***********************************************************************
***********************************************************************
      INTEGER FUNCTION NVIWX(FX,FY,FZ,X1,Y1,Z1,X3,IB,JB,KB,IE,JE,KE)
***********************************************************************
*USED ONLY IN SUBROUTINE VIEW

      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      COMMON/R4/XC(40),YC(40),ZC(40),XS(40),YS(40),ZS(40),DXXC(40),
     &          DYYC(40),DZZC(40),DXXS(40),DYYS(40),DZZS(40)

      NVIWX=0
      TPARA=(X3-X1)/(FX-X1)
      Y3=(FY-Y1)*TPARA+Y1
      Z3=(FZ-Z1)*TPARA+Z1
      IF (Y3.LE.YS(JE).AND.Y3.GE.YS(JB)) THEN
         IF (Z3.LE.ZS(KE).AND.Z3.GE.ZS(KB)) NVIWX=1
      ENDIF

      RETURN
      END
```

```
*****************************************************************************
*****************************************************************************
      INTEGER FUNCTION NVIWY(FX,FY,FZ,X1,Y1,Z1,Y3,IB,JB,KB,IE,JE,KE)
*****************************************************************************
*USED ONLY IN SUBROUTINE VIEW

      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      COMMON/R4/XC(40),YC(40),ZC(40),XS(40),YS(40),ZS(40),DXXC(40),
     &          DYYC(40),DZZC(40),DXXS(40),DYYS(40),DZZS(40)

      NVIWY=0
      TPARA=(Y3-Y1)/(FY-Y1)
      X3=(FX-X1)*TPARA+X1
      Z3=(FZ-Z1)*TPARA+Z1
      IF (X3.LE.XS(IE).AND.X3.GE.XS(IB)) THEN
         IF (Z3.LE.ZS(KE).AND.Z3.GE.ZS(KB)) NVIWY=1
      ENDIF

      RETURN
      END

*****************************************************************************
*****************************************************************************
      INTEGER FUNCTION NVIWZ(FX,FY,FZ,X1,Y1,Z1,Z3,IB,JB,KB,IE,JE,KE)
*****************************************************************************
*USED ONLY IN SUBROUTINE VIEW

      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      COMMON/R4/XC(40),YC(40),ZC(40),XS(40),YS(40),ZS(40),DXXC(40),
     &          DYYC(40),DZZC(40),DXXS(40),DYYS(40),DZZS(40)

      NVIWZ=0
      TPARA=(Z3-Z1)/(FZ-Z1)
      Y3=(FY-Y1)*TPARA+Y1
      X3=(FX-X1)*TPARA+X1
      IF (Y3.LE.YS(JE).AND.Y3.GE.YS(JB)) THEN
         IF (X3.LE.XS(IE).AND.X3.GE.XS(IB)) NVIWZ=1
      ENDIF

      RETURN
      END

*****************************************************************************
*****************************************************************************
      FUNCTION SILIN(V1,V2,D1,D2)
*****************************************************************************
* SINGLE LINEAR INTERPOLATION

      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      SILIN=(V1*D2+V2*D1)/(D1+D2)

      RETURN
      END
```

LIST OF REFERENCES

1. Patankar, S.V., *Numerical Heat Transfer and Fluid Flow*, Hemisphere Publishing Company, New York, NY, 1980.

2. Department of Aerospace and Mechanical Engineering, University of Notre Dame, South Bend, IN, Technical Report, TR-37191-74-4, *"A Numerical Model for the Prediction of Two Dimensional Unsteady Flows of Multicomponent Gases with Strong Buoyancy Effect and Recirculation,"* by Michael L. Doria, November, 1974.

3. Doria, M.L., A.C. Ku., and J.R. Lloyd, "Numerical Modeling of Unsteady Flows Generated by Fire in a Corridor," *Sixteenth Symposium (International) on Combustion, The Combustion Institute*, pp. 1373-1383, 1977.

4. Satoh, K., K.T. Yang, J.R. Lloyd, and A.M. Kanbury, "Modeling of Turbulent Buoyant Flows in Aircraft Cabins," *Combustion Science and Technology*, vol. 39, pp. 107-118, 1984.

5. Yang, H.Q., K.T. Yang, and J.R. Lloyd, "Flow Transition in Laminar Flow in a Three-Dimensional Tilted Rectangular Enclosure," *Heat Transfer, vol. 4,* pp. 1495-1500, 1986.

6. Yang, H.Q., K.T. Yang, and J.R. Lloyd, "Laminar Natural Convection Flow Transition in Tilted Three-Dimensional Longitudinal Rectangular Enclosures," *Int. J. Heat Mass Transfer*, vol. 30, no. 8., pp. 1637-1644, 1987.

7. Satoh, K., J.R. Lloyd, K.T. Yang, and A.M. Kanbury, "Numerical Finite-Difference Study of the Oscillatory Behaviour of Vertically Vented Compartments," *Numerical Properties and Methodologies in Heat Transfer*, pp. 517-528, 1985.

8. Yang, K.T., and J.R. Lloyd, "Turboulent Buoyant Flow in Vented Simple and Complex Enclosures," *Natural Convection Fundamentals and Applications*, pp.303-329, 1985.

9. Yang, H.Q., J.R. Lloyd, and K.T. Yang, "Laminar Natural Convection Flow Transition in Tilted Three Dimensional Longitudinal Rectangular Enclosures", *International Journal of Heat and Mass Transfer*, vol. 30,no.8 pp. 1637-1644, 1987.

10. Chan, A.M.C., and S. Banerjee, "Three-Dimensional Numerical Analysis of Transient Natural Convection in Rectangular Enclosure," *Intl. J. Heat and Mass Transfer*, vol. 26, pp.1427-1438, 1983.

11. Nicollette, V.F., K.T. Yang, and J.R. Lloyd, "Transient Cooling by Natural Convection in a Two Dimensional Square Enclosure," *Intl. J. Heat Transfer*, vol. 28, no. 9, pp. 1721-1732, 1985.

12. Leonard, S.P., "A Convectively Stable , Third-Order Accurate Finite-Difference Method for Steady Two-Dimensional Flow and Heat Transfer," *Numerical Properties and Methodologies in Heat Transfer*, ed. T.M. Shih, Hemisphere Publishing Corp., Washington, D.C., pp. 211-226, 1983.

13. Lin T.F., C. Liu, and C.K. Chen, "Unsteady Buoyancy Induced Flow and Heat Transfer in a Heated Inclined Open Channel,"*Proceedings of the Ninth International Heat Transfer Conference*, vol. 2, no. 5-NC-8, pp. 525-530, 1990.

14. Minkowycz, W.J., E.M. Sparrow, G.L. Sneider, R.H. Pletcher, *Handbook of Numerical Heat Transfer*, John Wiley and Sons, Inc., New York, 1988.

15. Greenspan, D., and V.Cassulli, *Numerical Analysis For Applied Mathematics, Science, and Engineering,* Addison-Wesley Publishing Company, Menlo Park, CA, 1988.

16. Smith, G.D, *Numerical Solutions of Partial Differential Equations: Finite Difference Methods*, Oxford University Press, New York, 1985.

17. Schlichting, H., *Boundary Layer Theory*, 7th ed., Mcgraw-Hill, New York, 1979.

18. Sparrow, E.M., and R.D. Cess, *Radiation Heat Transfer*, Hemisphere Publishing Corporation, Washington, D.C., 1978.

19. *Magazine and Propellant Fire Tests*, NAWC TP 7128 (Draft), 1992

20. Nies, G.F., *Numerical Field Model Simulation of Full Scale Tests in a Closed Vessel*, Master's and Mechanical Engineer's Thesis, Naval Postgraduate School, Monterey, CA, December 1986.

21. Raycraft, J.K., *Numerical Field Model Simulation of Full Scale Fire Tests in a Closed Sperical/Cylindrical Vessel*,

Master's and Mechanical Engineer's Thesis, Naval Postgraduate School, Monterey, CA, December 1987.

22. Houck, R.R., *Numerical Field Model Simulation of Full Scale Fire Tests in a Closed Sperical/Cylindrical Vessel with Internal Ventilation*, Master's Thesis, Naval Postgraduate School, Monterey, CA, December 1988.

23. MCarthy, Timothy G., *Numerical Field Model Simulation of Full Scale Fire Tests in a Closed Spherical/ Cylindrical Vessel Using Advanced Computer Graphics Techniques*, Master's Thesis, Naval Postgraduate School, Monterey, CA, September 1991.

24. Thorkildsen, Kenneth J., *Numerical Field Model Simulation of Fire and Heat Transfer in a Rectangular Compartment*, Master's Thesis, Naval Postgraduate School, Monterey, CA, September 1992

INITIAL DISTRIBUTION LIST

No. Copies

1. Defense Technical Information Center — 2
   Cameron Station
   Alexandria, VA 22304-6145

2. Library, Code 52 — 2
   Naval Postgraduate School
   Monterey, CA 93943-5002

3. Chairman, Code ME — 1
   Department of Mechanical Engineering
   Naval Postgraduate School
   Monterey. CA 93943-5000

4. Naval Engineering Curricular Office — 1
   Code 34
   Naval Postgraduate School
   Monterey, CA 93943-5000

5. Professor M.D. Kelleher, Code ME/kk — 2
   Departement of Mechanical Engineering
   Naval Postgraduate School
   Monterey, CA 93943-5000

6. Dr. Patricia Tatum — 1
   Code 6183
   Naval Research Laboratory
   Washington D.C. 20375

7. Mr. James Gagorik — 1
   Code 211, ONT
   Balston Towers #1
   800 N. Quincy
   Arlington, VA 22217

8. Professor K.T. Yang — 1
   Department of Aerospace and
     Mechanical Engineering
   University of Notre Dame
   Notre Dame, IN 46556

9. Mr. Kent Farmer — 1
   Code 38903, Bldg 31618
   Naval Air Warfare Center
   China Lake, CA 93555

10. Michael A. Delaney 1
166 Hicks Street
Brooklyn, NY 11201

Thesis
D26363 Delaney
c.1 Numerical field model
simulation of full scale
fire tests in a closed
and an open compartment.